



Open Source Used In AppDynamics_OTIS_Pipeline 22.10.683

Cisco Systems, Inc.

www.cisco.com

Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.

Text Part Number: 78EE117C99-1444728867

This document contains licenses and notices for open source software used in this product. With respect to the free/open source software listed in this document, if you have any questions or wish to receive a copy of any source code to which you may be entitled under the applicable free/open source license(s) (such as the GNU Lesser/General Public License), please contact us at external-opensource-requests@cisco.com.

In your requests please include the following reference number 78EE117C99-1444728867

Contents

1.1 google-uuid v.1.1.2

1.1.1 Available under license

1.2 spf13-cast v1.4.1

1.2.1 Available under license

1.3 prometheus-common v0.32.1

1.3.1 Available under license

1.4 apache-http-client 4.5.13

1.4.1 Available under license

1.5 jetbrains-annotations 13.0

1.5.1 Available under license

1.6 smartystreets-assertions v1.2.1

1.6.1 Available under license

1.7 jackson-bom 2.12.5

1.7.1 Available under license

1.8 aws-java-sdk-::-services-::-aws-glue 2.17.52

1.8.1 Available under license

1.9 commons-codec 1.15

1.9.1 Available under license

1.10 javax-annotation-api 1.3.2

1.10.1 Available under license

1.11 docker-java 3.2.13

1.11.1 Available under license

1.12 caffeine-cache 3.0.3

1.12.1 Available under license

1.13 amazon-kinesis-client-library-for-java 2.3.9

1.13.1 Available under license

- 1.14 slf4j-api-module 1.7.36**
 - 1.14.1 Available under license
- 1.15 aws-java-sdk-http-clients-apache 2.17.52**
 - 1.15.1 Available under license
- 1.16 apache-avro 1.10.2**
 - 1.16.1 Available under license
- 1.17 pkgerrors v0.9.1**
 - 1.17.1 Available under license
- 1.18 objenesis 3.2**
 - 1.18.1 Available under license
- 1.19 google-go-cmp v0.5.6**
 - 1.19.1 Available under license
- 1.20 cenk/backoff v4.1.0**
 - 1.20.1 Available under license
- 1.21 x-sys 20220114-snapshot-da31bd32**
 - 1.21.1 Available under license
- 1.22 go-logrus v1.8.1**
 - 1.22.1 Available under license
- 1.23 classgraph 4.8.120**
 - 1.23.1 Available under license
- 1.24 aws-java-sdk-::-http-clients-::-url-connection 2.17.52**
 - 1.24.1 Available under license
- 1.25 protoc-gen-validate v0.6.1**
 - 1.25.1 Available under license
- 1.26 kotlin-reflect 1.3.50**
 - 1.26.1 Available under license
- 1.27 jackson-annotations 2.13.0**
 - 1.27.1 Available under license
- 1.28 jackson-bom 2.13.0**
 - 1.28.1 Available under license
- 1.29 guava 30.1-android**
 - 1.29.1 Available under license
- 1.30 cespare-xxhash v2.1.2**
 - 1.30.1 Available under license
- 1.31 jmes-path-query-library 1.12.26**
 - 1.31.1 Available under license
- 1.32 asm-tree 9.1**
 - 1.32.1 Available under license
- 1.33 aws-java-sdk-utilities 2.17.52**

- 1.33.1 Available under license
- 1.34 guava-internalfuturefailureaccess-and-internalfutures 1.0.1**
 - 1.34.1 Available under license
- 1.35 junit-5-bill-of-materials 5.8.2**
 - 1.35.1 Available under license
- 1.36 apache-commons-digester 2.1**
 - 1.36.1 Available under license
- 1.37 go-tomb-tomb 20180513-snapshot-d5d1b582**
 - 1.37.1 Available under license
- 1.38 jtolds-gls v4.20.0**
 - 1.38.1 Available under license
- 1.39 commons-compress 1.21**
 - 1.39.1 Available under license
- 1.40 jackson-dataformats-binary 2.13.0**
 - 1.40.1 Available under license
- 1.41 client-golang v1.12.0**
 - 1.41.1 Available under license
- 1.42 pgv-java-stubs 0.6.1**
 - 1.42.1 Available under license
- 1.43 apache-commons-collections 3.2.2**
 - 1.43.1 Available under license
- 1.44 netty-codec-redis 4.1.72.Final**
 - 1.44.1 Available under license
- 1.45 netty-project 4.1.68.Final**
 - 1.45.1 Available under license
- 1.46 burntsushi-toml v0.3.1**
 - 1.46.1 Available under license
- 1.47 jackson-core 2.12.5**
 - 1.47.1 Available under license
- 1.48 protobuf v1.27.1**
 - 1.48.1 Available under license
- 1.49 aws-java-sdk-::third-party-::jackson-dataformat-cbor 2.17.52**
 - 1.49.1 Available under license
- 1.50 aws-java-sdk-for-amazon-dynamodb 1.12.26**
 - 1.50.1 Available under license
- 1.51 jackson-annotations jackson-annotations-2.12.3**
 - 1.51.1 Available under license
- 1.52 aws-java-sdk-::services-::amazon-cloudwatch 2.17.52**
 - 1.52.1 Available under license

1.53 aws-java-sdk---core---protocols---aws-cbor-protocol 2.17.52

1.53.1 Available under license

1.54 zstd-jni 1.4.9-1

1.54.1 Available under license

1.55 aws-java-sdk---services---amazon-dynamodb 2.17.52

1.55.1 Available under license

1.56 apache-log4j-api 2.17.1

1.56.1 Available under license

1.57 junit 4.13.1

1.57.1 Available under license

1.58 thanhpk/randstr v1.0.4

1.58.1 Available under license

1.59 aws-java-sdk-auth 2.17.52

1.59.1 Available under license

1.60 fsnotify-fsnotify v1.4.9

1.60.1 Available under license

1.61 checker-qual 3.5.0

1.61.1 Available under license

1.62 protocol-buffer-java-util-package 3.15.6

1.62.1 Available under license

1.63 io-grpc-grpc-protobuf 1.39.0

1.63.1 Available under license

1.64 aws-event-stream 1.0.1

1.64.1 Available under license

1.65 jackson-databind 2.12.5

1.65.1 Available under license

1.66 kotlin-scripting-common 1.3.50

1.66.1 Available under license

1.67 aws-glue-schema-registry-common 1.1.5

1.67.1 Available under license

1.68 netty-reactive-streams-implementation 2.0.5

1.68.1 Available under license

1.69 junit-jupiter-junit-jupiter-params 5.8.2

1.69.1 Available under license

1.70 perfmark-perfmark-api 0.23.0

1.70.1 Available under license

1.71 apache-commons-validator 1.7

1.71.1 Available under license

1.72 mockito v3.11.2

- 1.72.1 Available under license
- 1.73 j2objc-annotations 1.3**
 - 1.73.1 Available under license
- 1.74 lumberjack v2.0.0**
 - 1.74.1 Available under license
- 1.75 aws-java-sdk-core-protocols-protocol-core 2.17.52**
 - 1.75.1 Available under license
- 1.76 kotlin-script-runtime 1.3.50**
 - 1.76.1 Available under license
- 1.77 aws-java-sdk-annotations 2.17.52**
 - 1.77.1 Available under license
- 1.78 apache-log4j 2.17.1**
 - 1.78.1 Available under license
- 1.79 aws-java-sdk-secretsmanager 1.11.409**
 - 1.79.1 Available under license
- 1.80 snappy-java 1.1.8.1**
 - 1.80.1 Available under license
- 1.81 mac-os 4.1.72.Final**
 - 1.81.1 Available under license
- 1.82 io-grpc-grpc-testing 1.39.0**
 - 1.82.1 Available under license
- 1.83 jackson-dataformats-binary 2.12.3**
 - 1.83.1 Available under license
- 1.84 junit-platform-junit-platform-engine 1.8.2**
 - 1.84.1 Available under license
- 1.85 yaml-for-go v2.4.0**
 - 1.85.1 Available under license
- 1.86 reactive-streams v1.0.3**
 - 1.86.1 Available under license
- 1.87 checker-qual 3.15.0**
 - 1.87.1 Available under license
- 1.88 allegro-bigcache v2.2.5**
 - 1.88.1 Available under license
- 1.89 testcontainers--:-localstack 1.17.3**
 - 1.89.1 Available under license
- 1.90 joda-time v2.10.2**
 - 1.90.1 Available under license
- 1.91 aws-java-sdk--:-metrics-interface 2.17.52**
 - 1.91.1 Available under license

- 1.92 error_prone_annotations 2.3.4**
 - 1.92.1 Available under license
- 1.93 dgryski/go-rendezvous 20200823-snapshot-9f7001d1**
 - 1.93.1 Available under license
- 1.94 kotlin-scripting-jvm 1.3.50**
 - 1.94.1 Available under license
- 1.95 junit-jupiter-junit-jupiter-engine 5.8.2**
 - 1.95.1 Available under license
- 1.96 rxjava 2.1.14**
 - 1.96.1 Available under license
- 1.97 matttproud-golang-protobuf-extensions v1.0.1**
 - 1.97.1 Available under license
- 1.98 asm 9.1**
 - 1.98.1 Available under license
- 1.99 testcontainers-junit-jupiter-extension 1.17.3**
 - 1.99.1 Available under license
- 1.100 ginkgo 1.16.4**
 - 1.100.1 Available under license
- 1.101 docker-java-transport 3.2.13**
 - 1.101.1 Available under license
- 1.102 jackson-annotations 2.12.5**
 - 1.102.1 Available under license
- 1.103 junit-jupiter-junit-jupiter-api 5.8.2**
 - 1.103.1 Available under license
- 1.104 junit 4.12**
 - 1.104.1 Available under license
- 1.105 project-lombok 1.18.20**
 - 1.105.1 Available under license
- 1.106 jna 5.8.0**
 - 1.106.1 Available under license
- 1.107 x-sync 20210219-snapshot-036812b2**
 - 1.107.1 Available under license
- 1.108 pgv-java-grpc-interceptors 0.6.1**
 - 1.108.1 Available under license
- 1.109 golang-mock v1.6.0**
 - 1.109.1 Available under license
- 1.110 kotlin-stdlib-common 1.3.50**
 - 1.110.1 Available under license
- 1.111 prometheus-procfs v0.7.3**

- 1.111.1 Available under license
- 1.112 pgv-java-parent 0.6.1**
 - 1.112.1 Available under license
- 1.113 aws-java-sdk::-arns 2.17.52**
 - 1.113.1 Available under license
- 1.114 kotlin-scripting-compiler-impl-embeddable 1.3.50**
 - 1.114.1 Available under license
- 1.115 coocood-freecache v1.2.0**
 - 1.115.1 Available under license
- 1.116 jetbrains-kotlin-kotlin-scripting-compiler-embeddable 1.3.50**
 - 1.116.1 Available under license
- 1.117 goconvey v1.7.2**
 - 1.117.1 Available under license
- 1.118 slf4j-api-module 1.7.32**
 - 1.118.1 Available under license
- 1.119 netty/resolver/dns/classes/macos 4.1.72.Final**
 - 1.119.1 Available under license
- 1.120 go-humanize v1.0.0**
 - 1.120.1 Available under license
- 1.121 jackson-databind 2.13.0**
 - 1.121.1 Available under license
- 1.122 yaml-for-go 20200512-snapshot-9f266ea9**
 - 1.122.1 Available under license
- 1.123 aws-sdk-for-java-core 1.12.26**
 - 1.123.1 Available under license
- 1.124 go-spew v1.1.1**
 - 1.124.1 Available under license
- 1.125 netty-codec-smtp 4.1.72.Final**
 - 1.125.1 Available under license
- 1.126 aws-java-sdk::-third-party::-jackson-core 2.17.52**
 - 1.126.1 Available under license
- 1.127 pmezard-go-difflib 1.0.0**
 - 1.127.1 Available under license
- 1.128 error_prone_annotations 2.7.1**
 - 1.128.1 Available under license
- 1.129 apache-commons-lang 3.8.1**
 - 1.129.1 Available under license
- 1.130 kubernetes-apimachinery 20191123-snapshot-4c4803ed**
 - 1.130.1 Available under license

1.131 duct-tape 1.0.8

1.131.1 Available under license

1.132 grpc-netty-shaded 1.39.0

1.132.1 Available under license

1.133 google-api-grpc-proto-google-common-protos 2.0.1

1.133.1 Available under license

1.134 prometheus-client-model v0.2.0

1.134.1 Available under license

1.135 aws-java-sdk-aws-core 2.17.52

1.135.1 Available under license

1.136 protocol-buffers 3.11.4

1.136.1 Available under license

1.137 byte-buddy-agent 1.11.3

1.137.1 Available under license

1.138 guava v30.0

1.138.1 Available under license

1.139 aws-sdk-for-java-core 1.12.29

1.139.1 Available under license

1.140 aws-java-sdk-regions 2.17.52

1.140.1 Available under license

1.141 junit-platform-junit-platform-commons 1.8.2

1.141.1 Available under license

1.142 beorn7-perks v1.0.1

1.142.1 Available under license

1.143 junit 4.13.2

1.143.1 Available under license

1.144 netty/transport/classes/kqueue 4.1.72.Final

1.144.1 Available under license

1.145 google-gson 2.8.6

1.145.1 Available under license

1.146 aws-java-sdk-http-client-interface 2.17.52

1.146.1 Available under license

1.147 go-tomb-tomb 20150422-snapshot-dd632973

1.147.1 Available under license

1.148 dgraph-io/ristretto v0.1.0

1.148.1 Available under license

1.149 protocol-buffers 3.19.3

1.149.1 Available under license

1.150 gomemcache 20190913-snapshot-a41fca85

- 1.150.1 Available under license
- 1.151 apache-commons-digester 1.8.1**
 - 1.151.1 Available under license
- 1.152 aws-sdk-for-go v1.40.54**
 - 1.152.1 Available under license
- 1.153 assertj-fluent-assertions 3.22.0**
 - 1.153.1 Available under license
- 1.154 cespare-xxhash v1.1.0**
 - 1.154.1 Available under license
- 1.155 x-xerrors 20200804-snapshot-5ec99f83**
 - 1.155.1 Available under license
- 1.156 apache-log4j-slf4j-binding 2.17.1**
 - 1.156.1 Available under license
- 1.157 golang-glog 20160125-snapshot-23def4e6**
 - 1.157.1 Available under license
- 1.158 aws-java-sdk-sdk-core 2.17.52**
 - 1.158.1 Available under license
- 1.159 mockito-inline 3.11.2**
 - 1.159.1 Available under license
- 1.160 go-check-check 20190902-snapshot-41f04d3b**
 - 1.160.1 Available under license
- 1.161 hamcrest 1.3**
 - 1.161.1 Available under license
- 1.162 error_prone_annotations 2.4.0**
 - 1.162.1 Available under license
- 1.163 jetbrains-annotations 17.0.0**
 - 1.163.1 Available under license
- 1.164 project-lombok 1.18.12**
 - 1.164.1 Available under license
- 1.165 apache-commons-validator 1.6**
 - 1.165.1 Available under license
- 1.166 jackson-core 2.12.3**
 - 1.166.1 Available under license
- 1.167 apiguardian-apiguardian-api 1.1.2**
 - 1.167.1 Available under license
- 1.168 jackson-dataformats-binary 2.12.5**
 - 1.168.1 Available under license
- 1.169 tail v1.4.8**
 - 1.169.1 Available under license

- 1.170 junit-jupiter-aggregator 5.8.2**
 - 1.170.1 Available under license
- 1.171 io-grpc-grpc-core 1.39.0**
 - 1.171.1 Available under license
- 1.172 go-testify 1.7.0**
 - 1.172.1 Available under license
- 1.173 jmes-path-query-library 1.12.29**
 - 1.173.1 Available under license
- 1.174 docker-java-transport-zero-dep 3.2.13**
 - 1.174.1 Available under license
- 1.175 asm-analysis 9.1**
 - 1.175.1 Available under license
- 1.176 testcontainers-core 1.17.3**
 - 1.176.1 Available under license
- 1.177 bean-validation-api 2.0.1**
 - 1.177.1 Available under license
- 1.178 scala v2.12.10**
 - 1.178.1 Available under license
- 1.179 leaktest v1.3.0**
 - 1.179.1 Available under license
- 1.180 netty-transport-native-unix-common 4.1.72.Final**
 - 1.180.1 Available under license
- 1.181 byte-buddy byte-buddy-1.11.3**
 - 1.181.1 Available under license
- 1.182 patrickmn-go-cache 2.1.0**
 - 1.182.1 Available under license
- 1.183 software-amazon-ion-ion-java 1.0.2**
 - 1.183.1 Available under license
- 1.184 re2j 1.3**
 - 1.184.1 Available under license
- 1.185 apache-kafka 2.8.1**
 - 1.185.1 Available under license
- 1.186 aws-secretsmanager-caching-java 1.0.1**
 - 1.186.1 Available under license
- 1.187 golang-protobuf v1.5.2**
 - 1.187.1 Available under license
- 1.188 opentest4j-opentest4j 1.2.0**
 - 1.188.1 Available under license
- 1.189 kotlin 1.3.50**

- 1.189.1 Available under license
- 1.190 kotlinox-coroutines-core 1.1.1**
 - 1.190.1 Available under license
- 1.191 gomega v1.16.0**
 - 1.191.1 Available under license
- 1.192 guava-listenablefuture-only 9999.0-empty-to-avoid-conflict-with-guava**
 - 1.192.1 Available under license
- 1.193 commons-logging 1.2**
 - 1.193.1 Available under license
- 1.194 mbknor-jackson-jsonschema 1.0.39**
 - 1.194.1 Available under license
- 1.195 pegasus-kv/thrift v0.13.0**
 - 1.195.1 Available under license
- 1.196 lz4-and-xxhash 1.7.1**
 - 1.196.1 Available under license
- 1.197 aws-java-sdk-for-aws-sts 1.12.29**
 - 1.197.1 Available under license
- 1.198 animal-sniffer-annotation 1.19**
 - 1.198.1 Available under license
- 1.199 x-net 20210928-snapshot-e81a3d93**
 - 1.199.1 Available under license
- 1.200 google-android-annotations-library 4.1.1.4**
 - 1.200.1 Available under license
- 1.201 jackson-databind 2.12.3**
 - 1.201.1 Available under license
- 1.202 aws-java-sdk-::-core-::-protocols-::-json-utils 2.17.52**
 - 1.202.1 Available under license
- 1.203 aws-java-sdk-core-protocols-aws-query-protocol 2.17.52**
 - 1.203.1 Available under license
- 1.204 checker-qual 2.5.5**
 - 1.204.1 Available under license
- 1.205 opencensus-api 0.28.0**
 - 1.205.1 Available under license
- 1.206 io-grpc-grpc-stub 1.39.0**
 - 1.206.1 Available under license
- 1.207 netty/transport/classes/epoll 4.1.72.Final**
 - 1.207.1 Available under license
- 1.208 apache-httpcomponents-core 4.4.13**
 - 1.208.1 Available under license

- 1.209 aws-java-sdk-profiles 2.17.52**
 - 1.209.1 Available under license
- 1.210 json-java 20201115**
 - 1.210.1 Available under license
- 1.211 gopherjs 20181205-snapshot-0766667c**
 - 1.211.1 Available under license
- 1.212 mockito-junit-jupiter 3.11.2**
 - 1.212.1 Available under license
- 1.213 apache-avro 1.11.0**
 - 1.213.1 Available under license
- 1.214 netty-project 4.1.72.Final**
 - 1.214.1 Available under license
- 1.215 grpc-go v1.40.0**
 - 1.215.1 Available under license
- 1.216 findbugs-jsr305 3.0.2**
 - 1.216.1 Available under license
- 1.217 netty-transport-native-unix-common 4.1.68.Final**
 - 1.217.1 Available under license
- 1.218 everit-json-schema 1.12.2**
 - 1.218.1 Available under license
- 1.219 io-grpc-grpc-api 1.39.0**
 - 1.219.1 Available under license
- 1.220 slf4j-api-module 1.7.30**
 - 1.220.1 Available under license
- 1.221 io-grpc-grpc-protobuf-lite 1.39.0**
 - 1.221.1 Available under license
- 1.222 jackson-core 2.13.0**
 - 1.222.1 Available under license
- 1.223 io-grpc-grpc-context 1.39.0**
 - 1.223.1 Available under license
- 1.224 jmespath-go-jmespath v0.4.0**
 - 1.224.1 Available under license
- 1.225 jacoco 0.8.7**
 - 1.225.1 Available under license
- 1.226 godoc-text v0.1.0**
 - 1.226.1 Available under license
- 1.227 aws-java-sdk-for-aws-kms 1.12.26**
 - 1.227.1 Available under license
- 1.228 aws-java-sdk-for-amazon-s3 1.12.26**

- 1.228.1 Available under license
- 1.229 kr-pretty 0.1.0**
 - 1.229.1 Available under license
- 1.230 asm-commons 9.1**
 - 1.230.1 Available under license
- 1.231 aws-java-sdk-::-core-::-protocols-::-aws-json-protocol 2.17.52**
 - 1.231.1 Available under license
- 1.232 text v0.3.7**
 - 1.232.1 Available under license
- 1.233 handy-uri-templates handy-uri-templates-2.1.8**
 - 1.233.1 Available under license
- 1.234 netty-reactive-streams-http-support 2.0.5**
 - 1.234.1 Available under license
- 1.235 github.com/xiaomi/pegasus-go-client 20210427-snapshot-f3b6b08b**
 - 1.235.1 Available under license
- 1.236 re2j 1.5**
 - 1.236.1 Available under license
- 1.237 aws-java-sdk-::-services-::-amazon-kinesis 2.17.52**
 - 1.237.1 Available under license
- 1.238 aws-java-sdk-http-clients-netty-non-blocking-i-o 2.17.52**
 - 1.238.1 Available under license
- 1.239 golang-appengine v1.6.7**
 - 1.239.1 Available under license
- 1.240 netty/tomcatnative-[openssl---classes] 2.0.46.Final**
 - 1.240.1 Available under license
- 1.241 kqueue 4.1.72.Final**
 - 1.241.1 Available under license
- 1.242 aws-glue-schema-registry-avro-serializer-deserializer 1.1.5**
 - 1.242.1 Available under license
- 1.243 vmihailenco/msgpack v4.0.4**
 - 1.243.1 Available under license
- 1.244 aws-glue-schema-registry-build-tools 1.1.5**
 - 1.244.1 Available under license

1.1 google-uuid v.1.1.2

1.1.1 Available under license :

Paul Borman <borman@google.com>

bmatsuo

shawnps

theory

jboverfelt
dsymonds
cd1
wallclockbuilder
dansouza

Copyright (c) 2009,2014 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2 spf13-cast v1.4.1

1.2.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Steve Francia

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.3 prometheus-common v0.32.1

1.3.1 Available under license :

Common libraries shared by Prometheus Go components.
Copyright 2015 The Prometheus Authors

This product includes software developed at
SoundCloud Ltd. (<http://soundcloud.com/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.4 apache-http-client 4.5.13

1.4.1 Available under license :

Apache HttpComponents Client
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

=====

This project includes Public Suffix List copied from
<https://publicsuffix.org/list/effective_tld_names.dat>
licensed under the terms of the Mozilla Public License, v. 2.0

Full license text: <<http://mozilla.org/MPL/2.0/>>

Mozilla Public License Version 2.0

=====

1. Definitions

1.1. "Contributor"

means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"

means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

means Covered Software of a particular Contributor.

1.4. "Covered Software"

means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"

means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"

means any form of the work other than Source Code Form.

1.7. "Larger Work"

means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"

means this document.

1.9. "Licensable"

means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"

means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor

means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its

Contributor Version.

1.12. "Secondary License"

means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"

means the form of the work preferred for making modifications.

1.14. "You" (or "Your")

means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

- (a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and
- (b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under

this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

- (a) for any code that a Contributor has removed from Covered Software;
or
- (b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
- (c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

- (a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and
- (b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support,

indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all

end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

* *
* 6. Disclaimer of Warranty *
* ----- *
* *
* Covered Software is provided under this License on an "as is" *
* basis, without warranty of any kind, either expressed, implied, or *
* statutory, including, without limitation, warranties that the *
* Covered Software is free of defects, merchantable, fit for a *
* particular purpose or non-infringing. The entire risk as to the *
* quality and performance of the Covered Software is with You. *
* Should any Covered Software prove defective in any respect, You *
* (not any Contributor) assume the cost of any necessary servicing, *
* repair, or correction. This disclaimer of warranty constitutes an *
* essential part of this License. No use of any Covered Software is *
* authorized under this License except under this disclaimer. *
* *

* *
* 7. Limitation of Liability *
* ----- *
* *
* Under no circumstances and under no legal theory, whether tort *
* (including negligence), contract, or otherwise, shall any *
* Contributor, or anyone who distributes Covered Software as *
* permitted above, be liable to You for any direct, indirect, *
* special, incidental, or consequential damages of any character *
* including, without limitation, damages for lost profits, loss of *
* goodwill, work stoppage, computer failure or malfunction, or any *
* and all other commercial damages or losses, even if such party *
* shall have been informed of the possibility of such damages. This *
* limitation of liability shall not apply to liability for death or *
* personal injury resulting from such party's negligence to the *
* extent applicable law prohibits such limitation. Some *
* jurisdictions do not allow the exclusion or limitation of *
* incidental or consequential damages, so this exclusion and *
* limitation may not apply to You. *
* *

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the

notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

1.5 jetbrains-annotations 13.0

1.5.1 Available under license :

Copyright 2000-2012 JetBrains s.r.o.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.6 smartystreets-assertions v1.2.1

1.6.1 Available under license :

```
# This is the official list of people who can contribute
# (and typically have contributed) code to the go-diff
# repository.
#
```

The AUTHORS file lists the copyright holders; this file
lists people. For example, ACME Inc. employees would be listed here
but not in AUTHORS, because ACME Inc. would hold the copyright.

When adding J Random Contributor's name to this file,
either J's name or J's organization's name should be
added to the AUTHORS file.

Names should be added to this file like so:
Name <email address>

Please keep the list sorted.

Danny Yoo <dannyoo@google.com>
James Kolb <jkolb@google.com>
Jonathan Amsterdam <jba@google.com>
Markus Zimmermann <markus.zimmermann@nethead.at> <markus.zimmermann@symflower.com>
<zimmski@gmail.com>
Matt Kovars <akaskik@gmail.com>
rjan Persson <orjan@spotify.com>
Osman Masood <oamasood@gmail.com>
Robert Carlsen <rwcarlsen@gmail.com>
Rory Flynn <rorlyflynn@users.noreply.github.com>
Sergi Mansilla <sergi.mansilla@gmail.com>
Shatrugna Sadhu <ssadhu@apcera.com>
Shawn Smith <shawnpsmith@gmail.com>
Stas Maksimov <maksimov@gmail.com>
Tor Arvid Lund <torarvid@gmail.com>
Zac Bergquist <zbergquist99@gmail.com>
Copyright (c) 2012-2016 The go-diff Authors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2016 SmartyStreets, LLC

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

NOTE: Various optional and subordinate components carry their own licensing requirements and restrictions. Use of those components is subject to the terms and conditions outlined the respective license of each component.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

```
// Copyright (c) 2015 The Chromium Authors. All rights reserved.  
//  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions are  
// met:  
//  
// * Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// * Redistributions in binary form must reproduce the above  
// copyright notice, this list of conditions and the following disclaimer  
// in the documentation and/or other materials provided with the  
// distribution.  
// * Neither the name of Google Inc. nor the names of its  
// contributors may be used to endorse or promote products derived from  
// this software without specific prior written permission.
```

//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.7 jackson-bom 2.12.5

1.7.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.8 aws-java-sdk-::-services-::-aws-glue

2.17.52

1.8.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by
Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.9 commons-codec 1.15

1.9.1 Available under license :

Apache Commons Codec
Copyright 2002-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

`src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java`
contains test data from <http://aspell.net/test/orig/batch0.tab>.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

=====

The content of package `org.apache.commons.codec.language.bm` has been translated
from the original php source code available at <http://stevemorse.org/phoneticinfo.htm>
with permission from the original authors.

Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.10 javax-annotation-api 1.3.2

1.10.1 Available under license :

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor

either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or

alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms

herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY.S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original

Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or

binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that

accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception statement from your version.

1.11 docker-java 3.2.13

1.11.1 Available under license :

Apache-2.0

1.12 caffeine-cache 3.0.3

1.12.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/MpscGrowableArrayQueue.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2014 Ben Manes. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
```

- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Ticker.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/stats/CacheStats.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/LinkedDeque.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Policy.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/RemovalListener.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/stats/DisabledStatsCounter.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Weigher.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/AsyncLoadingCache.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/BoundedLocalCache.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Cache.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/RemovalCause.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WriteOrderDeque.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/UnboundedLocalCache.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/AccessOrderDeque.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/stats/StatsCounter.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/stats/ConcurrentStatsCounter.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Caffeine.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-

jar/com/github/benmanes/caffeine/cache/LoadingCache.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-

jar/com/github/benmanes/caffeine/cache/AbstractLinkedDeque.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-

jar/com/github/benmanes/caffeine/cache/CacheLoader.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2015 Ben Manes. All Rights Reserved.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

/*

* Written by Doug Lea with assistance from members of JCP JSR-166

* Expert Group and released to the public domain, as explained at

* <http://creativecommons.org/publicdomain/zero/1.0/>

*/

Found in path(s):

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-

jar/com/github/benmanes/caffeine/cache/StripedBuffer.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017 Ben Manes. All Rights Reserved.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Expiry.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/TimerWheel.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2018 Ben Manes. All Rights Reserved.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/AsyncCache.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/LocalAsyncCache.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 Ben Manes. All Rights Reserved.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

```
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/AsyncCacheLoader.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/CaffeineSpec.java
No license file was found, but licenses were detected in source scan.
```

```
// Copyright 2021 Ben Manes. All Rights Reserved.
// Licensed under the Apache License, Version 2.0 (the "License");
// you may not use this file except in compliance with the License.
// You may obtain a copy of the License at
// http://www.apache.org/licenses/LICENSE-2.0
// distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):

```
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/SSLMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/SSLSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/WILMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/WSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/WILMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/WLSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/SSMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/WILW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/SSLSMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/WSLMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/SIMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/SIS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/WIMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/WILSMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/FWAWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/SISAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-
jar/com/github/benmanes/caffeine/cache/FWAW.java
```

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSARMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWAWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSAWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSARMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDARMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDAWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDRMS.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDARMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWAW.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PD.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDAWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSARMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDAMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLSMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMSW.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSAWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDAWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSAWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDAMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMWR.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSAWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDAWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMWAR.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDAWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSAWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLSMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWARMW.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDAWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWAMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSL.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMRAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWAWR.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSAWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDARMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSARMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMSWR.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDAWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSL.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLSMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWAWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSAWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLR.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIL.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWAWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMWA.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDAMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSLW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDARMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWAWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSLAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/LocalCacheFactory.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMSAW.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSAMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMSR.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSAWR.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FD.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SI.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WI.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWMW.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWARMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWAMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMSAWR.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMRAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSWRMS.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWAMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSAMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSAMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWARMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSLR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWAMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSWMW.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSAWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIL.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWAWRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSAMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMWAW.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FDAMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WISMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/NodeFactory.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWAWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SISMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMSA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMSAW.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWARMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMWAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLSMWWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDAWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDAWR.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSLMWW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SILMWA.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PDWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMSW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSRMS.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILMSWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WIMWAR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FWWRMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/PWAWMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSLSMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMSA.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SIMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FSR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMSAW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSMWAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSMSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WILSMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SSSAWR.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WSSMW.java
* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WLSMSR.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2019 Ben Manes. All Rights Reserved.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Scheduler.java

* /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Pacer.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2015 Ben Manes. All Rights Reserved.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/FrequencySketch.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Buffer.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Async.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/LocalAsyncLoadingCache.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/References.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/stats/package-info.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/LocalLoadingCache.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/WriteThroughEntry.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/Node.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/BoundedBuffer.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/package-info.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/SerializationProxy.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/LocalCache.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/stats/GuardedStatsCounter.java
- * /opt/cola/permits/1341639961_1654809995.3473003/0/caffeine-3-0-3-sources-jar/com/github/benmanes/caffeine/cache/LocalManualCache.java

1.13 amazon-kinesis-client-library-for-java

2.3.9

1.13.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<!--  
/*  
* Copyright 2019 Amazon.com, Inc. or its affiliates.  
* Licensed under the Apache License, Version 2.0 (the  
* "License"); you may not use this file except in compliance  
* with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/  
-->
```

Found in path(s):

```
*/opt/cola/permits/1411867094_1662685304.1152492/0/amazon-kinesis-client-2-3-9-jar/META-INF/maven/software.amazon.kinesis/amazon-kinesis-client/pom.xml
```

1.14 slf4j-api-module 1.7.36

1.14.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/**  
* Copyright (c) 2004-2011 QOS.ch  
* All rights reserved.  
*  
* Permission is hereby granted, free of charge, to any person obtaining  
* a copy of this software and associated documentation files (the  
* "Software"), to deal in the Software without restriction, including  
* without limitation the rights to use, copy, modify, merge, publish,  
* distribute, sublicense, and/or sell copies of the Software, and to  
* permit persons to whom the Software is furnished to do so, subject to  
* the following conditions:  
*  
* The above copyright notice and this permission notice shall be  
* included in all copies or substantial portions of the Software.  
*  
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,  
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF  
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
```

* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
* LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
* OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*
*/

Found in path(s):

* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/FormattingTuple.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/NOPLoggerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/spi/MDCAdapter.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/MarkerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/impl/StaticLoggerBinder.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/impl/StaticMarkerBinder.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/MDC.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/MessageFormatter.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/Marker.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/BasicMDCAdapter.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/BasicMarker.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/ILoggerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/impl/StaticMDCBinder.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/MarkerIgnoringBase.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/NOPMDCAdapter.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/NamedLoggerBase.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/BasicMarkerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/Util.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/LoggerFactory.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/spi/LoggerFactoryBinder.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/spi/LocationAwareLogger.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/helpers/SubstituteLogger.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/Logger.java
* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-

jar/org/slf4j/spi/MarkerFactoryBinder.java

* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-

jar/org/slf4j/helpers/NOPLogger.java

* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-jar/org/slf4j/IMarkerFactory.java

* /opt/cola/permits/1331474007_1653510300.1759446/0/slf4j-api-1-7-36-sources-

jar/org/slf4j/helpers/SubstituteLoggerFactory.java

1.15 aws-java-sdk-http-clients-apache 2.17.52

1.15.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:

Apache Commons Lang

Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.16 apache-avro 1.10.2

1.16.1 Available under license :

Apache Avro
Copyright 2009-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.17 pkgerrors v0.9.1

1.17.1 Available under license :

Copyright (c) 2015, Dave Cheney <dave@cheney.net>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR

SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.18 objenesis 3.2

1.18.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate

as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify

the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include

the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.19 google-go-cmp v0.5.6

1.19.1 Available under license :

Copyright (c) 2017 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.20 cenk/backoff v4.1.0

1.20.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Cenk Alt

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.21 x-sys 20220114-snapshot-da31bd32

1.21.1 Available under license :

```
# This source code was written by the Go contributors.  
# The master list of contributors is in the main Go distribution,  
# visible at http://tip.golang.org/CONTRIBUTORS.  
Copyright (c) 2009 The Go Authors. All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.22 go-logrus v1.8.1

1.22.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Simon Eskildsen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.23 classgraph 4.8.120

1.23.1 Available under license :

No license file was found, but licenses were detected in source scan.

<name>The MIT License (MIT)</name>

Found in path(s):

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/META-

INF/maven/io.github.classgraph/classgraph/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* This file is part of ClassGraph.

*

* Author: Michael J. Simons

*

* Hosted at: <https://github.com/classgraph/classgraph>

*

* --

*

* The MIT License (MIT)

*

* Copyright (c) 2019 Luke Hutchison

*

* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated

* documentation files (the "Software"), to deal in the Software without restriction, including without

* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of

* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following

* conditions:

*

* The above copyright notice and this permission notice shall be included in all copies or substantial

* portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT

* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO

* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN

* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE

* OR OTHER DEALINGS IN THE SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-

jar/nonapi/io.github.classgraph/classloaderhandler/SpringBootRestartClassLoaderHandler.java

No license file was found, but licenses were detected in source scan.


```
/*
 * This file is part of ClassGraph.
 *
 * Author: Sergey Beshpalov
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --
 *
 * The MIT License (MIT)
 *
 * Copyright (c) 2017 Sergey Beshpalov
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
 * documentation files (the "Software"), to deal in the Software without restriction, including without
 * limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
 * the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
 * conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies or substantial
 * portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT
 * LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
 * AND NONINFRINGEMENT. IN NO
 * EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
 * OR OTHER LIABILITY, WHETHER IN
 * AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE
 * OR OTHER DEALINGS IN THE SOFTWARE.
 */
```

Found in path(s):

```
*/opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-
jar/nonapi/io/github/classgraph/classloaderhandler/WebSphereTraditionalClassLoaderHandler.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * This file is part of ClassGraph.
 *
 * Author: @mcollovati
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --
 *
 * The MIT License (MIT)
```

*
* Copyright (c) 2019 @mcollovati, contributed to the ClassGraph project
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
* AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
* OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-
jar/nonapi/io/github/classgraph/classloaderhandler/QuarkusClassLoaderHandler.java
No license file was found, but licenses were detected in source scan.

/*
* This file is part of ClassGraph.
*
* Author: Luke Hutchison
*
* Hosted at: <https://github.com/classgraph/classgraph>
*
* --
*
* The MIT License (MIT)
*
* Copyright (c) 2021 Luke Hutchison
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial

* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-
jar/nonapi/io/github/classgraph/classloaderhandler/CxfContainerClassLoaderHandler.java
No license file was found, but licenses were detected in source scan.

/*
* This file is part of ClassGraph.
*
* Author: Luke Hutchison (luke.hutch@gmail.com)
*
* Hosted at: <https://github.com/classgraph/classgraph>
*
* --
*
* The MIT License (MIT)
*
* Copyright (c) 2019 Luke Hutchison
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE

* OR OTHER DEALINGS IN THE SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ResourceList.java

No license file was found, but licenses were detected in source scan.

/*

* This file is part of ClassGraph.

*

* Author: Johno Crawford (johno@sulake.com)

*

* Hosted at: <https://github.com/classgraph/classgraph>

*

* --

*

* The MIT License (MIT)

*

* Copyright (c) 2016 Johno Crawford

*

* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated

* documentation files (the "Software"), to deal in the Software without restriction, including without

* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of

* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following

* conditions:

*

* The above copyright notice and this permission notice shall be included in all copies or substantial

* portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT

* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO

* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN

* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE

* OR OTHER DEALINGS IN THE SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/concurrency/AutoCloseableExecutorService.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/concurrency/SimpleThreadFactory.java

No license file was found, but licenses were detected in source scan.

```
/*
 * This file is part of ClassGraph.
 *
 * Author: Luke Hutchison
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --
 *
 * The MIT License (MIT)
 *
 * Copyright (c) 2019 Luke Hutchison, with significant contributions from Davy De Durlpel
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
 * documentation files (the "Software"), to deal in the Software without restriction, including without
 * limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
 * the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
 * conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies or substantial
 * portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT
 * LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
 * AND NONINFRINGEMENT. IN NO
 * EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
 * OR OTHER LIABILITY, WHETHER IN
 * AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE
 * OR OTHER DEALINGS IN THE SOFTWARE.
 */
```

Found in path(s):

```
*/opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-
jar/nonapi/io/github/classgraph/classloaderhandler/JBossClassLoaderHandler.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * This file is part of ClassGraph.
 *
 * Author: R. Kempees
 *
 * With contributions from @cpierceworld (#414)
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --
```

*
* The MIT License (MIT)
*
* Copyright (c) 2017 R. Kempees (contributed to the ClassGraph project)
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
* AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
* OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-
jar/nonapi/io/github/classgraph/classloaderhandler/WebSphereLibertyClassLoaderHandler.java
No license file was found, but licenses were detected in source scan.

/*
* This file is part of ClassGraph.
*
* Author: Harith Elrifaie
*
* Hosted at: <https://github.com/classgraph/classgraph>
*
* --
*
* The MIT License (MIT)
*
* Copyright (c) 2017 Harith Elrifaie
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:

*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-
jar/nonapi/io/github/classgraph/classloaderhandler/FelixClassLoaderHandler.java
No license file was found, but licenses were detected in source scan.

/*
* This file is part of ClassGraph.
*
* Author: Luke Hutchison
*
* Hosted at: <https://github.com/classgraph/classgraph>
*
* --
*
* The MIT License (MIT)
*
* Copyright (c) 2019 Luke Hutchison
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
OR OTHER LIABILITY, WHETHER IN

* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/AnnotationClassRef.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/OSGiDefaultClassLoaderHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ModulePathInfo.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/FieldInfo.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClassInfoList.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/EquinoxContextFinderClassLoaderHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/recycler/RecycleOnClose.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classpath/ClasspathOrder.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/JSONUtils.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClassRefTypeSignature.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/ClassFieldCache.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/concurrency/SingletonMap.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/PotentiallyUnmodifiableList.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/urls/URLPathEncoder.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/types/Parser.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/types/TypeUtils.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/BaseTypeSignature.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/FallbackClassLoaderHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/ParameterizedTypeImpl.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/urls/FileUtils.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/urls/LogNode.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/AnnotationParameterValue.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/GraphvizDotfileGenerator.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/TypeVariableSignature.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClassTypeSignature.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/TypeParameter.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/FieldTypeInfo.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classpath/ModuleFinder.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/TomcatWebappClassLoaderBaseHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ScanResultObject.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/ClassLoaderHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClasspathElementFileDir.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/AnnotationEnumValue.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/scanspec/ScanSpec.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/Utils/StringUtils.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/PackageInfoList.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classpath/CallStackReader.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/MappableInfoList.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ModuleInfo.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/JSONArray.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/AnnotationInfo.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClasspathElementZip.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/Utils/ReflectionUtils.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/Scanner.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/Utils/VersionFinder.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/ClassGraphClassLoaderHandler.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/JSONObject.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/Utils/FastPathResolver.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/concurrency/InterruptChecker.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClassInfo.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/MethodTypeSignature.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/PackageInfo.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/JPMSCClassLoaderHandler.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/concurrency/WorkQueue.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ScanResult.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClassGraphClassLoader.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/TypeResolutions.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/Utils/JarUtils.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/AnnotationInfoList.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ModuleReaderProxy.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fastzipfilereader/FastZipEntry.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ArrayClassInfo.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ModuleRef.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClassGraphException.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/HierarchicalTypeSignature.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClasspathElement.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ArrayTypeSignature.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/Classfile.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/recycler/Resettable.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/JSONReference.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/HasName.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/WeblogicClassLoaderHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/ReferenceEqualityKey.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/JSONParser.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/Resource.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/TypeArgument.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/ParentLastDelegationOrderTestClassLoaderHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classpath/ClassLoaderFinder.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClasspathElementPathDir.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/PlexusClassWorldsClassRealmClassLoaderHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClassGraph.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/ClassFields.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/EquinoxClassLoaderHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/URLClassLoaderHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/InfoList.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ReferenceTypeSignature.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/types/ParseException.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/recycler/Recycler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fastzipfilereader/LogicalZipFile.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClassRefOrTypeVariableSignature.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/ClassLoaderHandlerRegistry.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/JSONDeserializer.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/MethodInfoList.java

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/Id.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/MethodInfo.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/TypeSignature.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ObjectTypedValueWrapper.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/json/JSONSerializer.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classloaderhandler/UnoOneJarClassLoaderHandler.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/AnnotationParameterValueList.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ModuleInfoList.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classpath/SystemJarFinder.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/FieldInfoList.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fastzipfilereader/PhysicalZipFile.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/Utils/CollectionUtils.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classpath/ClassLoaderOrder.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/MethodParameterInfo.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/scanspec/AcceptReject.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/io/github/classgraph/ClasspathElementModule.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/classpath/ClasspathFinder.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fastzipfilereader/ZipFileSlice.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fastzipfilereader/NestedJarHandler.java
No license file was found, but licenses were detected in source scan.

/*

* This file is part of ClassGraph.

*

* Author: Luke Hutchison

*

* Hosted at: <https://github.com/classgraph/classgraph>

*

* --

*
* The MIT License (MIT)
*
* Copyright (c) 2020 Luke Hutchison
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
* AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
* OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fileslice/Slice.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fileslice/ArraySlice.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fileslice/reader/SequentialReader.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fileslice/reader/RandomAccessArrayReader.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fileslice/reader/RandomAccessByteBufferReader.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fileslice/reader/ClassfileReader.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fileslice/reader/RandomAccessFileChannelReader.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fileslice/reader/RandomAccessReader.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fileslice/FileSlice.java
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-jar/nonapi/io/github/classgraph/fileslice/PathSlice.java

No license file was found, but licenses were detected in source scan.

```
/*
 * This file is part of ClassGraph.
 *
 * Author: @jacobg on GitHub
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --
 *
 * The MIT License (MIT)
 *
 * Copyright (c) 2019 @jacobg, Luke Hutchison
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
 * documentation files (the "Software"), to deal in the Software without restriction, including without
 * limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
 * the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
 * conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies or substantial
 * portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT
 * LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
 * AND NONINFRINGEMENT. IN NO
 * EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
 * OR OTHER LIABILITY, WHETHER IN
 * AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE
 * OR OTHER DEALINGS IN THE SOFTWARE.
 */
```

Found in path(s):

```
* /opt/cola/permits/1411867118_1662683845.1356816/0/classgraph-4-8-120-sources-
jar/nonapi/io/github/classgraph/classloaderhandler/AntClassLoaderHandler.java
```

1.24 aws-java-sdk-::-http-clients-::-url-connection 2.17.52

1.24.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:

Apache Commons Lang

Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<https://www.apache.org/>).

1.25 protoc-gen-validate v0.6.1

1.25.1 Available under license :

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.26 kotlin-reflect 1.3.50

1.26.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2010-2018 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
*/opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/renderer/KeywordStringsGenerated.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```


*/

Found in path(s):

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/storage/StorageManager.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/VariableDescriptorWithAccessors.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/DisjointKeysUnionTypeSubstitution.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/descriptors/DeserializedMemberScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/incremental/components/LookupTracker.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/descriptors/JavaMethodDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/KProperty0Impl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/AnnotationUseSiteTarget.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/KTypeImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/VariableDescriptorWithInitializerImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/ParameterDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/components/DescriptorResolverUtils.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/VariableDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/incremental/components/LookupLocation.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/AnnotatedImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/AbstractScopeAdapter.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/KotlinClassFinder.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/MemberDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/name/SpecialNames.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/name/Name.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/TransientReceiver.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/PropertyGetterDescriptorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/internal/EmptyContainerForLocal.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/pcollections/IntTree.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/builtins/functions/BuiltInFictitiousFunctionClassFactory.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/KClassImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/ReflectProperties.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/AnnotationArgumentVisitor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/SimpleFunctionDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/AnnotatedCallableKind.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/components/ReflectAnnotationSource.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/PropertyDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/ModuleClassResolver.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/components/JavaResolverCache.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/PackageViewDescriptor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/descriptors/JavaClassDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/AnnotationDescriptorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/structure/annotationArguments.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/Receiver.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/ModuleDescriptor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/components/RuntimeErrorReporter.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/InnerClassesScopeWrapper.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/util.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/LazyPackageViewDescriptorImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaElement.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/AnnotationAndConstantLoaderImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/internal/impl/load/java/FieldOverridabilityCondition.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/CompositePackageFragmentProvider.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/DeclarationDescriptorVisitorEmptyBodies.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/propertiesConventionUtil.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/PackageFragmentProviderImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/DeclarationDescriptorWithVisibility.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/VariableDescriptorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/KotlinJvmBinaryPackageSourceElement.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/reflectLambda.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/incremental/Utils.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/builtins/DefaultBuiltIns.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/JvmAnnotationNames.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/ResolutionScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/ClassDescriptorBase.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/SubstitutingScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/PropertyGetterDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/KProperty2Impl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/LazyClassReceiverParameterDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaPackageFragment.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/CastImplicitClassReceiver.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/DescriptorSubstitutor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/name/FqNamesUtil.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/ModuleDescriptorImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/JavaClassDataFinder.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/internal/impl/util/scopeUtils.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/builtins/functions/FunctionClassScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/StaticScopeForKotlinEnum.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/ClassOrPackageFragmentDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/TypeProjection.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/KotlinJvmBinaryClass.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/KDeclarationContainerImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/utils/capitalizeDecapitalize.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/Named.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/JavaDescriptorUtil.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/BinaryClassAnnotationAndConstantLoaderImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/ErrorHandler.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/AbstractReceiverParameterDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/ClassDeserializer.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/DeserializedClassDataFinder.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/utils/collections.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/name/FqNameUnsafe.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/descriptors/JavaCallableMemberDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/DeclaredMemberIndex.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/name/FqName.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/DeclarationDescriptorWithSource.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/ClassKind.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/AbstractBinaryClassAnnotationAndConstantLoader.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/storage/storage.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/internal/impl/descriptors/ValueParameterDescriptor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/KotlinRetention.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/VariableAccessorDescriptor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/resolvers.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/PropertyAccessorDescriptorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/ImplicitClassReceiver.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/util/ModuleVisibilityHelper.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/builtins/CompanionObjectMapping.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/pcollections/IntTreePMap.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/VisibilityUtil.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/ChainedMemberScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/ReflectionObjectRenderer.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/header/KotlinClassHeader.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/AnnotationDescriptor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/typeParameterUtils.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/KCallablesJvm.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/ProtoEnumFlags.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/sources/JavaSourceElementFactory.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/LazyScopeAdapter.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/utills/Printer.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/utills/WrappedValues.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/SimpleFunctionDescriptorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/moduleByClassLoader.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/kClassCache.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/internal/KProperty1Impl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/PropertyAccessorDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/pcollections/HashPMap.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/constants/constantValues.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/ImplicitReceiver.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/ExtensionReceiver.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/KFunctionImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/ReceiverValue.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/MemberComparator.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/LazyJavaPackageFragmentProvider.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/DeclarationDescriptorVisitor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/impl/exceptionUtils.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/AbstractLazyTypeParameterDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/ReceiverParameterDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/Variance.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/JavaIncompatibilityRulesOverridabilityCondition.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/descriptors/DeserializedPackageMemberScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/components/SignaturePropagator.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/ReceiverParameterDescriptorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/impl/impl/addToStdlib.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/ReflectJvmMapping.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/ValueParameterDescriptorImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/SupertypeLoopChecker.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/jvm/JavaDescriptorResolver.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/internal/impl/load/java/components/JavaAnnotationMapper.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/descriptors/JavaPropertyDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/ClassTypeConstructorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/DescriptorFactory.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/utils/SmartSet.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/DeclarationDescriptorNonRootImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/DeserializedPackageFragment.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/pcollections/MapEntry.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/AnnotationWithTarget.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/KotlinJvmBinarySourceElement.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/AnnotationAndConstantLoader.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/TypeProjectionImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/Visibility.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/header/ReadKotlinClassHeaderAnnotationVisitor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/AbstractReceiverValue.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/FunctionWithAllInvokes.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/PackageFragmentDescriptor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/ExternalOverridabilityCondition.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/CallableMemberDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/DeserializedDescriptorResolver.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/KPackageImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/LocalClassifierTypeSettings.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/DeclarationDescriptorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/specialBuiltinMembers.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/internal/impl/resolve/jvm/JvmClassName.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaModifierListOwner.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/descriptors/DeserializedAnnotations.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/components/TypeUsage.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/SubpackagesScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/renderer/RenderingUtils.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/checker/TypeCheckerProcedureCallbacksImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/builtins/functions/FunctionInvokeDescriptor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/MemberScopeImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/descriptors/util.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/utills/coreLib.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/storage/NoLock.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/DeclarationDescriptorNonRoot.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/MemberScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/StarProjectionImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/PackageFragmentDescriptorImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/TypeConstructor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/ReflectionFactoryImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/ClassDataFinder.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/utills/strings.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/name/ClassId.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/annotationUtil.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/JvmAnnotationNames.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/CallableDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/KClassesJvm.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/ConstUtil.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/PropertySetterDescriptorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaStaticScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/RuntimeTypeMapper.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/CompositionTypeSubstitution.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/KParameterImpl.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/JavaClassFinder.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/checker/TypeCheckingProcedureCallbacks.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/EffectiveVisibility.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/storage/ObservableStorageManager.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/DeclarationDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/SourceElement.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/jvm/JvmPrimitiveType.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/LazyJavaAnnotations.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/pcollections/ConsPStack.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/JavaVisibilities.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/FakePureImplementationsProvider.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/PackageFragmentProvider.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/TypeParameterDescriptorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/TypeParameterDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/SourceFile.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/ProtoContainer.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/builtins/PrimitiveType.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

```
jar/kotlin/reflect/jvm/internal/impl/resolve/constants/ConstantValueFactory.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/serialization/SerializerExtensionProtocol.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/ScriptDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/JvmPackagePartSource.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/overridingUtils.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/PropertySetterDescriptor.java
No license file was found, but licenses were detected in source scan.
```

```
/*
* Copyright 2010-2015 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
// A seemingly obvious way to come about this case would be to declare a special exception class, but the problem is
that
```

```
Found in path(s):
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/storage/LockBasedStorageManager.java
No license file was found, but licenses were detected in source scan.
```

```
/*
* Copyright 2010-2017 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/components/ReflectKotlinClassFinder.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/checker/IntersectionType.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/PropertyDescriptorImpl.java
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/SyntheticScopes.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/SpecialTypes.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/components/RuntimeSourceElementFactory.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/structure/ReflectJavaAnnotation.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/deserialization/PlatformDependentDeclarationFilter.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/Substitutable.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/structure/ReflectJavaValueParameter.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/builtins/BuiltInsLoader.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/checker/NewKotlinTypeChecker.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/TypeWithEnhancement.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/structure/ReflectJavaAnnotationOwner.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/JavaClassesTracker.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/DeserializedPackageFragmentImpl.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaScope.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/descriptors/JavaClassConstructorDescriptor.java
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/Visibilities.java
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/structure/ReflectJavaClassifierType.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/ContractDeserializer.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaPackageScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/name/NameUtils.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/deserialization/AdditionalClassPartsProvider.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaType.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/utils.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaClassMemberScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/DescriptorDerivedFromTypeAlias.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaArrayType.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaMethod.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaWildcardType.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/components/SamConversionResolver.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/resolve/OverridingUtil.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaPackage.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/TypeUtils.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/findClassInModule.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/incremental/components/ExpectActualTracker.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/container/DefaultImplementation.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/builtins/BuiltInsPackageFragment.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/AnnotationTypeQualifierResolver.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/checker/NewCapturedType.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/components/ReflectJavaClassFinder.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaMember.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/typeEnhancement/signatureEnhancement.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/utils/numbers.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/internal/impl/descriptors/deserialization/ClassDescriptorFactory.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/KotlinTypeFactory.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/ConstructorDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaAnnotationArguments.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/reflectClassUtil.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/utills/Jsr305State.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/full/KProperties.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/BuiltInAnnotationDescriptor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/KotlinType.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaConstructor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaClass.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaPrimitiveType.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaField.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/util/OperatorNameConventions.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/TypeAliasConstructorDescriptor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/ClassConstructorDescriptorImpl.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/util/modifierChecks.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/components/RuntimeModuleData.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/config/KotlinCompilerVersion.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/structure/ReflectJavaTypeParameter.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/components/ReflectKotlinClass.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/descriptors/InvalidModuleException.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/types/ErrorType.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaStaticClassScope.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

jar/kotlin/reflect/jvm/internal/impl/descriptors/Modality.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2010-2016 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/Utils/SmartList.java

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/ClassifierDescriptor.java

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/IncompatibleVersionErrorData.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/ErasedOverridabilityCondition.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/ClassifierDescriptorWithTypeParameters.java

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/KTypesJvm.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/full/exceptions.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/IntersectionTypeConstructor.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/resolve/constants/CompileTimeConstant.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/full/KClasses.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/SuperCallReceiverValue.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/TypeAliasDescriptor.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/types/JavaTypeResolver.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/structure/javaTypes.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-

- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/methodSignatureBuilding.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/typeEnhancement/typeEnhancement.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/EmptyPackageFragmentDescriptor.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/TypeProjectionBase.java
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/ErrorUtils.java
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/KTypeParameterImpl.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/AbstractClassDescriptor.java
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/TypeCapabilities.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/AnnotationDeserializer.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/KotlinReflectionInternalError.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/JavaFlexibleTypeDeserializer.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/resolve/constants/IntegerValueTypeConstructor.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/MetadataPackageFragmentProvider.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/full/KClassifiers.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/checker/Utils.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/GivenFunctionsMemberScope.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/structure/javaElements.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/MemberSignature.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/AbstractTypeConstructor.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaAnnotationDescriptor.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/typeEnhancement/typeQualifiers.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/renderer/DescriptorRendererOptionsImpl.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/TypeSubstitution.kt
- * /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/Utils/functions.kt

* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/AbstractClassTypeConstructor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/FunctionPlaceholders.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/types/checker/ClassicTypeCheckerContext.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/resolve/DescriptorEquivalenceForOverrides.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/typeEnhancement/predefinedEnhancementInfo.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/renderer/ClassifierNamePolicy.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/context.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/resolve/calls/inference/CapturedTypeConstructor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/AbstractTypeParameterDescriptor.java
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/Annotations.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/load/java/components/JavaPropertyInitializerEvaluator.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/ClassConstructorDescriptor.kt
* /opt/cola/permits/1136558942_1613789270.78/0/kotlin-reflect-1-3-50-sources-3-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/AbstractTypeAliasDescriptor.kt

1.27 jackson-annotations 2.13.0

1.27.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or

documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,

work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.28 jackson-bom 2.13.0

1.28.1 Available under license :

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>

<url><http://www.apache.org/licenses/LICENSE-2.0.txt></url>

Found in path(s):

* /opt/cola/permits/1340798855_1654756534.4125829/0/jackson-bom-2-12-3-pom-1-zip/jackson-bom-2.12.3.pom

1.29 guava 30.1-android

1.29.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2020 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/ServiceManagerBridge.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Java8Usage.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/GwtTransient.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2008 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/EmptyImmutableListMultimap.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/RegularImmutableBiMap.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/StandardTable.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/TreeBasedTable.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableCollection.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/CollectPreconditions.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/HashBasedTable.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/Table.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableListMultimap.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/RegularImmutableMap.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableBiMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/thirdparty/publicsuffix/PublicSuffixPatterns.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/UnmodifiableIterator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/StandardRowSortedTable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/Serialization.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableMapKeySet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableMapEntrySet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/Platform.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/PeekingIterator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableSortedSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/Range.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableEntry.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableMapValues.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/Tables.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/Collections2.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableMultimap.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/SortedMultisets.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/SortedMultiset.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2019 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/CompactHashing.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/IncidentEdgeSet.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2014 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/eventbus/Dispatcher.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/MoreObjects.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/math/Quantiles.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/eventbus/Subscriber.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/ListenerCallQueue.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/TrustedListenableFutureTask.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/eventbus/SubscriberRegistry.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2013 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Utf8.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/FilteredMultimapValues.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/thirdparty/publicsuffix/PublicSuffixType.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/eventbus/SubscriberExceptionContext.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/eventbus/SubscriberExceptionHandler.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Verify.java

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/WrappingScheduledExecutorService.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/Runnables.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/VerifyException.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/TypeVisitor.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/CharSequenceReader.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/HashingInputStream.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractTable.java
```

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2016 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/NetworkBuilder.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/DirectedGraphConnections.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/AbstractUndirectedNetworkConnections.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/AbstractNetwork.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ImmutableMultisetGwtSerializationDependencies.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/MapIteratorCache.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/ValueGraphBuilder.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
```

jar/com/google/common/graph/UndirectedGraphConnections.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/StandardNetwork.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/ValueGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/StandardMutableValueGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/RangeGwtSerializationDependencies.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/ForwardingValueGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/ForwardingGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/HashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/StandardMutableNetwork.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/UndirectedNetworkConnections.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/UndirectedMultiNetworkConnections.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/StandardMutableGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/GraphConnections.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/DirectedNetworkConnections.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/ElementOrder.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/EdgesConnecting.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/GraphConstants.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/ImmutableValueGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/MapRetrievalCache.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/AbstractDirectedNetworkConnections.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/DirectedMultiNetworkConnections.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ArrayListMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/MultiEdgesConnecting.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/NetworkConnections.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/graph/GraphBuilder.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/EndpointPair.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/MutableValueGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/ForwardingNetwork.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/AbstractValueGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/EndpointPairIterator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Comparators.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/AbstractGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/StandardValueGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/LinkedHashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/AbstractGraphBuilder.java
No license file was found, but licenses were detected in source scan.

```
/*  
 * Copyright (C) 2012 The Guava Authors  
 *  
 * Licensed under the Apache License, Version 2.0 (the "License");  
 * you may not use this file except in compliance with the License.  
 * You may obtain a copy of the License at  
 *  
 * http://www.apache.org/licenses/LICENSE-2.0  
 *  
 * Unless required by applicable law or agreed to in writing, software  
 * distributed under the License is distributed on an "AS IS" BASIS,  
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
 * See the License for the specific language governing permissions and  
 * limitations under the License.  
 */
```

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/CompactLinkedHashSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingNavigableSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/SortedMultisetBridge.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/DescendingMultiset.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ForwardingDeque.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/CompactHashSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/EvictingQueue.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ForwardingNavigableMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/AbstractSortedKeySortedSetMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ForwardingImmutableMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/AbstractMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/AbstractNavigableMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/FilteredMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/TreeRangeMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/CompactHashMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/FilteredKeyListMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/FilteredEntrySetMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/TransformedIterator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/AllEqualOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/RegularImmutableAsList.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ForwardingBlockingDeque.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/ForwardingBlockingDeque.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ForwardingImmutableSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/CompactLinkedHashMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/FilteredSetMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/FilteredEntryMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ForwardingImmutableList.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/FilteredKeySetMultimap.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableEnumMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/TreeTraverser.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/RangeMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/UnmodifiableSortedMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/DescendingImmutableSortedSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/TransformedListIterator.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableEnumSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/EmptyImmutableSetMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/SingletonImmutableTable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableAsList.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ComparisonChain.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/RegularImmutableSortedSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableTable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ComputationException.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/collect/ArrayTable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingTable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ImmutableClassToInstanceMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/RegularImmutableList.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ImmutableSortedSetFauxverideShim.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ImmutableSortedMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ImmutableSetMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractIndexedListIterator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/DiscreteDomain.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2017 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ClosingFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/AbstractBaseGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ObjectCountHashMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/Traverser.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ObjectCountLinkedHashMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/BaseGraph.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2006 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/TimeoutFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AbstractCatchingFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/TypeToken.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/PatternFilenameFilter.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/GwtFluentFutureCatchingSpecialization.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ImmediateFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/annotations/VisibleForTesting.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/UncheckedTimeoutException.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/CollectionFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/escape/CharEscaperBuilder.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/AppendableWriter.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AggregateFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/FuturesGetChecked.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/CaseFormat.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/Futures.java
```

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/AbstractTransformFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/GwtFuturesCatchingSpecialization.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/SimpleTimeLimiter.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/TimeLimiter.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/FakeTimeLimiter.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/escape/CharEscaper.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/FluentFuture.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2010 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ForwardingSortedSetMultimap.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ForwardingSetMultimap.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/MinMaxPriorityQueue.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ForwardingListMultimap.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/RowSortedTable.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/SortedMapDifference.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/AbstractSequentialIterator.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/collect/UnmodifiableListIterator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingImmutableCollection.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2007 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except  
* in compliance with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software distributed under the License  
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either  
express  
* or implied. See the License for the specific language governing permissions and limitations under  
* the License.  
*/  
/*  
* This following method is a modified version of one found in  
* http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/test/tck/AbstractExecutorServiceTest.java?revision=1.30  
* which contained the following notice:  
*  
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to  
* the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/  
*  
* Other contributors include Andrew Wright, Jeffrey Hayes, Pat Fisher, Mike Judd.  
*/
```

Found in path(s):
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/MoreExecutors.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2019 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except  
* in compliance with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software distributed under the License  
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either  
express  
* or implied. See the License for the specific language governing permissions and limitations under  
* the License.
```

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/IgnoreJRERRequirement.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/CombinedFuture.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/hash/FarmHashFingerprint64.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/Platform.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/InterruptibleTask.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/AsyncCallable.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ConsumingQueueIterator.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/hash/MacHashFunction.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/AggregateFutureState.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/hash/LittleEndianByteArray.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/ReaderInputStream.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2016 The Guava Authors

*

```
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for extra methods of { @code Objects } only in web. Intended to be empty for regular
* version.
*/
```

Found in path(s):

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/ExtraObjectsMethodsForWeb.java
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Copyright (C) 2011 The Guava Authors
```

```
/*
```

```
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
```

```
/*
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

```
/*
```

```
* Unless required by applicable law or agreed to in writing, software distributed under the License
```

```
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
```

```
* or implied. See the License for the specific language governing permissions and limitations under
```

```
* the License.
```

```
*/
```

Found in path(s):

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/net/HttpHeaders.java
```

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/PrimitiveSink.java
```

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/Uninterruptibles.java
```

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/Absent.java
```

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/Optional.java
```

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
```

jar/com/google/common/cache/RemovalCause.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/DoubleMath.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/primitives/UnsignedInteger.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/IntMath.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/Cache.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/Present.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/primitives/UnsignedLongs.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/primitives/ParseRequest.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/AbstractLoadingCache.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/MathPreconditions.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/CacheStats.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/net/MediaType.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/Weigher.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/Enums.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/FunctionalEquivalence.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/AbstractNonStreamingHashFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/LongMath.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/FutureCallback.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/TreeRangeSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/Hashing.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/HashingOutputStream.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AbstractListeningExecutorService.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/Ticker.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/Funnel.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/DoubleUtils.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/Funnels.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/Murmur3_128HashFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ExecutionError.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/EmptyContiguousSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/Murmur3_32HashFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/net/HostAndPort.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Queues.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/RemovalListeners.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/ForwardingCache.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/primitives/UnsignedInts.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/BloomFilterStrategies.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AsyncFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/WrappingExecutorService.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/Types.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/CacheLoader.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/UncheckedExecutionException.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ListeningScheduledExecutorService.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AtomicLongMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/reflect/TypeParameter.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/HashCode.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AbstractScheduledService.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/BoundType.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ForwardingListeningExecutorService.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/RegularContiguousSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/AbstractStreamingHasher.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/RemovalListener.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/RegularImmutableMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/DescendingImmutableSortedMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/ForwardingLoadingCache.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/AbstractCache.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/Hasher.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/BloomFilter.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractSortedMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/Crc32cHashFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/primitives/UnsignedLong.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/MessageDigestHashFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/BigIntegerMath.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/HashFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/AbstractCompositeHashFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/CacheBuilderSpec.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/LoadingCache.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/RemovalNotification.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/hash/AbstractHasher.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2010 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/ListeningExecutorService.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/UncaughtExceptionHandler.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/annotations/package-info.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/ForwardingBlockingQueue.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/collect/SortedLists.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/annotations/Beta.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/Monitor.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/base/Ascii.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/ThreadFactoryBuilder.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/collect/ContiguousSet.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/net/package-info.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/base/Equivalence.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/Atomics.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/primitives/package-info.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/base/Strings.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2017 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/primitives/ImmutableLongArray.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/ForwardingLock.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/hash/AbstractHashFunction.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/ForwardingCondition.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/primitives/ImmutableDoubleArray.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/primitives/ImmutableIntArray.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2008 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/escape/UnicodeEscaper.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/FileBackedOutputStream.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/net/InetAddresses.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/Bytes.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/MultiReader.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/thirdparty/publicsuffix/TrieParser.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/Chars.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/escape/Escaper.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/ListenableFutureTask.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/internal/Finalizer.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/SequentialExecutor.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/net/PercentEscaper.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/Doubles.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/CharMatcher.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/Booleans.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/Shorts.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/Floats.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/FluentIterable.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/Ints.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Stopwatch.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/Longs.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Converter.java
- * /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Joiner.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/OverflowAvoidingLockSupport.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/Java8Compatibility.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/BigDecimalMath.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/Java8Compatibility.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/ToDoubleRounder.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2014 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
```

jar/com/google/common/collect/TopKSelector.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/PredecessorsFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/Graph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/ImmutableNetwork.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/Graphs.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/ImmutableGraph.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/MutableNetwork.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/SuccessorsFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/Network.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/MutableGraph.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 The Guava Authors.

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/package-info.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/graph/package-info.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2020 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

/**

* Holder for web specializations of methods of { @code Doubles }. Intended to be empty for regular
* version.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/primitives/DoublesMethodsForWeb.java

No license file was found, but licenses were detected in source scan.

/*

* Written by Doug Lea with assistance from members of JCP JSR-166

* Expert Group and released to the public domain, as explained at

* <http://creativecommons.org/publicdomain/zero/1.0/>

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/LongAdder.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/cache/LongAdder.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/Striped64.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/Striped64.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AtomicDoubleArray.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2019 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except  
* in compliance with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software distributed under the License  
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either  
express  
* or implied. See the License for the specific language governing permissions and limitations under  
* the License.  
*/
```

Found in path(s):
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/primitives/Platform.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2009 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except  
* in compliance with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software distributed under the License  
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either  
express  
* or implied. See the License for the specific language governing permissions and limitations under  
* the License.  
*/  
/**  
* Outer class that exists solely to let us write {@code Partially.GwtIncompatible} instead of plain  
* {@code GwtIncompatible}. This is more accurate for {@link Futures#catching}, which is available  
* under GWT but with a slightly different signature.  
*  
*/
```

```
* <p>We can't use { @code PartiallyGwtIncompatible } because then the GWT compiler wouldn't recognize
* it as a { @code GwtIncompatible } annotation. And for { @code Futures.catching }, we need the GWT
* compiler to autostrip the normal server method in order to expose the special, inherited GWT
* version.
*/
```

Found in path(s):

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/Partially.java
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Copyright (C) 2016 The Guava Authors
```

```
*
```

```
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
```

```
*
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

```
*
```

```
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
```

```
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
```

```
*/
```

Found in path(s):

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/CommonPattern.java
```

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/JdkPattern.java
```

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/PatternCompiler.java
```

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/CommonMatcher.java
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Copyright (C) 2020 The Guava Authors
```

```
*
```

```
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
```

```
*
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

```
*
```

```
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
```


* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for web specializations of methods of { @code Floats }. Intended to be empty for regular
* version.
*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/primitives/FloatsMethodsForWeb.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2008 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

/*

* This method was rewritten in Java from an intermediate step of the Murmur hash function in

* <http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp>, which contained the
* following header:

*

* MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author

* hereby disclaims copyright to this source code.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Hashing.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2018 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/ImmutableSupplier.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ExecutionSequencer.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2007 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Sets.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/LinkedHashMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingCollection.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingObject.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractSetMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/BiMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/collect/ForwardingMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/SetMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/CompoundOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/SingletonImmutableSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractListMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Multiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/LexicographicalOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ArrayListMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingSortedMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Lists.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/NullsFirstOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Iterators.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ByFunctionOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Maps.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ExplicitOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ConcurrentHashMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/MutableClassToInstanceMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingSortedSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/HashMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/MapDifference.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/EnumBiMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/UsingToStringOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Ordering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/collect/Interner.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Synchronized.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/EnumHashMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ImmutableList.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractMapBasedMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Multimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractMapEntry.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ReverseNaturalOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/NaturalOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Iterables.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingListIterator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingIterator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Multisets.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingConcurrentMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractBiMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/LinkedListMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Multimaps.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractMapBasedMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ReverseOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/LinkedHashMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/TreeMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/SortedSetMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ImmutableSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/collect/AbstractSortedSetMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/TreeMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ClassToInstanceMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ComparatorOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/HashMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingMapEntry.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/NullsLastOrdering.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingList.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/AbstractIterator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/RegularImmutableSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ForwardingQueue.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ListMultimap.java

No license file was found, but licenses were detected in source scan.

```
/*  
 * Copyright (C) 2020 The Guava Authors  
 *  
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except  
 * in compliance with the License. You may obtain a copy of the License at  
 *  
 * http://www.apache.org/licenses/LICENSE-2.0  
 *  
 * Unless required by applicable law or agreed to in writing, software distributed under the License  
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either  
 * express  
 * or implied. See the License for the specific language governing permissions and limitations under  
 * the License.  
 */  
/**  
 * Holder for web specializations of methods of { @code Ints }. Intended to be empty for regular  
 * version.  
 */
```

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/IntsMethodsForWeb.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2013 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/MultimapBuilder.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2018 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/IndexedImmutableSet.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/BaseImmutableMultimap.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/Preconditions.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/LittleEndianDataOutputStream.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Interners.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/Closeables.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AbstractFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ListenableFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/CharStreams.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/Defaults.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/Predicate.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/AbstractIterator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/eventbus/AllowConcurrentEvents.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/MultiInputStream.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/DirectExecutor.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/Function.java
```

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/LineReader.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Predicates.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/LineBuffer.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/FinalizableSoftReference.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/FinalizableWeakReference.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/LittleEndianDataInputStream.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/CountingInputStream.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Supplier.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/ByteStreams.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Functions.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Throwables.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/eventbus/EventBus.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Charsets.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/eventbus/AsyncEventBus.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/util/concurrent/ExecutionList.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/Files.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/EnumMultiset.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/HashBiMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/Resources.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/Flushables.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/FinalizableReference.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/io/CountingOutputStream.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Suppliers.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/eventbus/Subscribe.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/Objects.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/eventbus/DeadEvent.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/eventbus/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/FinalizablePhantomReference.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/FinalizableReferenceQueue.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/Primitives.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
either

* express or implied. See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ForwardingSortedMultiset.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/AbstractRangeSet.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/SortedIterable.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/GeneralRange.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/RangeSet.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/RegularImmutableSortedMultiset.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/collect/ImmutableSortedMultiset.java

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/SortedIterables.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ImmutableSortedMultisetFauxverideShim.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Count.java
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Copyright (C) 2007 The Guava Authors
```

```
*
```

```
* Licensed under the Apache License, Version 2.0 (the "License");
```

```
* you may not use this file except in compliance with the License.
```

```
* You may obtain a copy of the License at
```

```
*
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

```
*
```

```
* Unless required by applicable law or agreed to in writing, software
```

```
* distributed under the License is distributed on an "AS IS" BASIS,
```

```
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

```
* See the License for the specific language governing permissions and
```

```
* limitations under the License.
```

```
*/
```

```
/**
```

```
* Returns an array containing all of the elements in the specified collection. This method
```

```
* returns the elements in the order they are returned by the collection's iterator. The returned
```

```
* array is "safe" in that no references to it are maintained by the collection. The caller is
```

```
* thus free to modify the returned array.
```

```
*
```

```
* <p>This method assumes that the collection size doesn't change while the method is running.
```

```
*
```

```
* <p>TODO(kevinb): support concurrently modified collections?
```

```
*
```

```
* @param c the collection for which to return an array of elements
```

```
*/
```

Found in path(s):

```
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ObjectArrays.java
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Copyright (C) 2012 The Guava Authors
```

```
*
```

```
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
```

```
* in compliance with the License. You may obtain a copy of the License at
```

```
*
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

/*

* This method was rewritten in Java from an intermediate step of the Murmur hash function in
* <http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp>, which contained the
* following header:

*

* MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
* hereby disclaims copyright to this source code.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/SmallCharMatcher.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2009 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

/**

* Not supported. **You are attempting to create a map that may contain a non-{@code Comparable}**

* **key.** Proper calls will resolve to the version in {@code ImmutableSortedMap}, not this dummy

* version.

*

* @throws UnsupportedOperationException always

* @deprecated **Pass a key of type {@code Comparable} to use {@link**

* **ImmutableSortedMap#of(Comparable, Object).**

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/collect/ImmutableSortedMapFauxverideShim.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2009 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/LineProcessor.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/Cut.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/escape/ArrayBasedEscaperMap.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/escape/ArrayBasedCharEscaper.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AbstractExecutionThreadService.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AbstractIdleService.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/net/UrlEscapers.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/annotations/GwtIncompatible.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ForwardingFluentFuture.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/escape/Platform.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/MapMaker.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/AbstractService.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/CacheBuilder.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/xml/XmlEscapers.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/ForwardingFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/Callables.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/ByteArrayDataInput.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/net/InternetDomainName.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/primitives/UnsignedBytes.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/Service.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/primitives/SignedBytes.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/MapMakerInternalMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/base/Platform.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/html/HtmlEscapers.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/TypeResolver.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/ByteArrayDataOutput.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/ReferenceEntry.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/JdkFutureAdapters.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/escape/ArrayBasedUnicodeEscaper.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/RegularImmutableTable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/annotations/GwtCompatible.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ForwardingListenableFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/escape/Escapers.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/LocalCache.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/net/HostSpecifier.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/SparseImmutableTable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/DenseImmutableTable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/SettableFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/base/Splitter.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/io/ByteProcessor.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

/*

* This method was written by Doug Lea with assistance from members of JCP JSR-166 Expert Group

* and released to the public domain, as explained at

* <http://creativecommons.org/licenses/publicdomain>

*

* As of 2010/06/11, this method is identical to the (package private) hash method in OpenJDK 7's

* `java.util.HashMap` class.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/util/concurrent/Striped.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2005 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/reflect/Reflection.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2020 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

/**

* Holder for web specializations of methods of { @code Shorts }. Intended to be empty for regular
* version.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/primitives/ShortsMethodsForWeb.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2012 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

Found in path(s):

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-jar/com/google/common/base/StandardSystemProperty.java

* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/math/Stats.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/AbstractByteHasher.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/PairedStatsAccumulator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/ChecksumHashFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/SmoothRateLimiter.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ImmutableRangeSet.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/ByteSource.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/Invokable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/StatsAccumulator.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/FilteredKeyMultimap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/Parameter.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/TypeCapture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/SipHashFunction.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ServiceManager.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/LinearTransformation.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/LongAddables.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/RateLimiter.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/CharSource.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/Closer.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/util/concurrent/ListenableScheduledFuture.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/BaseEncoding.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/Element.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/xml/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-

jar/com/google/common/collect/CartesianList.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/escape/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/math/PairedStats.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/collect/ImmutableRangeMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/CharSink.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/ImmutableTypeToInstanceMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/TypeToInstanceMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/LongAddable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/MutableTypeToInstanceMap.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/ClassPath.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/html/package-info.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/ByteSink.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/reflect/AbstractInvocationHandler.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/cache/LongAddable.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/hash/LongAddables.java
* /opt/cola/permits/1125300210_1615713995.83/0/guava-30-1-android-sources-1-
jar/com/google/common/io/FileWriteMode.java

1.30 cespare-xxhash v2.1.2

1.30.1 Available under license :

Copyright (c) 2016 Caleb Spare

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.31 jmes-path-query-library 1.12.26

1.31.1 Available under license :

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>

Found in path(s):

* /opt/cola/permits/1444731873_1666051478.6883318/0/jmespath-java-1-12-26-jar/META-INF/maven/com.amazonaws/jmespath-java/pom.xml

1.32 asm-tree 9.1

1.32.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Bundle-DocURL: <http://asm.ow2.org>

Bundle-License: BSD-3-Clause;link=<https://asm.ow2.io/LICENSE.txt>

Bundle-ManifestVersion: 2

Bundle-Name: org.objectweb.asm.tree

Bundle-RequiredExecutionEnvironment: J2SE-1.5

Bundle-SymbolicName: org.objectweb.asm.tree

Bundle-Version: 9.1.0

Export-Package: org.objectweb.asm.tree;version="9.1";uses:="org.objectweb.asm"

Implementation-Title: Tree API of ASM, a very small and fast Java byte code manipulation framework

Implementation-Version: 9.1

Import-Package: org.objectweb.asm;version="[9.1,10)"

Module-Requires: org.objectweb.asm;transitive=true

Found in path(s):

* /opt/cola/permits/1183890441_1627493647.83/0/asm-tree-9-1-jar/META-INF/MANIFEST.MF

1.33 aws-java-sdk-utilities 2.17.52

1.33.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade

names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier

identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and

limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed

with the accompanying software if such terms are included in the directory containing the accompanying software.

Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by

Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:

Apache Commons Lang

Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<https://www.apache.org/>).

1.34 guava-internalfuturefailureaccess-and-internalfutures 1.0.1

1.34.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1130987386_1612872111.26/0/failureaccess-1-0-1-sources-
jar/com/google/common/util/concurrent/internal/InternalFutureFailureAccess.java
* /opt/cola/permits/1130987386_1612872111.26/0/failureaccess-1-0-1-sources-
jar/com/google/common/util/concurrent/internal/InternalFutures.java
```

1.35 junit-5-bill-of-materials 5.8.2

1.35.1 Available under license :

```
import java.io.File
import java.net.URI
```

```
data class License(val name: String, val url: URI, val headerFile: File)
Apache License
=====
```

```
_Version 2.0, January 2004_
_&lt;<https://www.apache.org/licenses/>&gt;_
```

```
### Terms and Conditions for use, reproduction, and distribution
```

```
#### 1. Definitions
```

```
License shall mean the terms and conditions for use, reproduction, and
```


distribution as defined by Sections 1 through 9 of this document.

Licensors shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensors for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensors or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensors for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensors and any individual or Legal Entity on behalf

of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * *(a)* You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * *(b)* You must cause any modified files to carry prominent notices stating that You changed the files; and
- * *(c)* You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * *(d)* If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of

the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or

any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and

* **b)** in the case of each subsequent Contributor:

* **i)** changes to the Program, and

* **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form

that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

* ****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium

customarily used for software exchange; and

* **b)*** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

* **i)*** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* **ii)*** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

* **iii)*** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

* **iv)*** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)*** it must be made available under this Agreement, or if the Program *(i)* is combined with other material in a separate file or files made available under a Secondary License, and *(ii)* the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* **b)*** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Apache License

=====

Version 2.0, January 2004

<<<https://www.apache.org/licenses/>>>>

Terms and Conditions for use, reproduction, and distribution

1. Definitions

License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration

files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where

such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
- * **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets `[]` replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same printed page as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Open Source Licenses

=====

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for
these subcomponents is subject to the terms and conditions of the
subcomponent's license, as noted in the LICENSE-<subcomponent>.md
files.

[[contributors]]

== Contributors

Browse the {junit5-repo}/graphs/contributors[current list of contributors] directly on GitHub.

1.36 apache-commons-digester 2.1

1.36.1 Available under license :

Apache Commons Digester

Copyright 2001-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.37 go-tomb-tomb 20180513-snapshot-d5d1b582

1.37.1 Available under license :

tomb - support for clean goroutine termination in Go.

Copyright (c) 2010-2011 - Gustavo Niemeyer <gustavo@niemeyer.net>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.38 jtolds-gls v4.20.0

1.38.1 Available under license :

Copyright (c) 2013, Space Monkey, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.39 commons-compress 1.21

1.39.1 Available under license :

Apache Commons Compress

Copyright 2002-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

The files in the package `org.apache.commons.compress.archivers.sevenz` were derived from the LZMA SDK, version 9.20 (C/ and CPP/7zip/), which has been placed in the public domain:

"LZMA SDK is placed in the public domain." (<http://www.7-zip.org/sdk.html>)

The test file `lbzip2_32767.bz2` has been copied from `libbzip2`'s source repository:

This program, "bzip2", the associated library "libbzip2", and all documentation, are copyright (C) 1996-2019 Julian R Seward. All rights reserved.

Redistribution and use in source and binary forms, with or without

modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org
Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all

other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.40 jackson-dataformats-binary 2.13.0

1.40.1 Available under license :

JacksonIon

Copyright 2012-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including

the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each

Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.41 client-golang v1.12.0

1.41.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Prometheus instrumentation library for Go applications

Copyright 2012-2015 The Prometheus Authors

This product includes software developed at
SoundCloud Ltd. (<http://soundcloud.com/>).

The following components are included in this product:

perks - a fork of <https://github.com/bmizerany/perks>
<https://github.com/beorn7/perks>
Copyright 2013-2015 Blake Mizerany, Bjrn Rabenstein
See <https://github.com/beorn7/perks/blob/master/README.md> for license details.

Go support for Protocol Buffers - Google's data interchange format
<http://github.com/golang/protobuf/>
Copyright 2010 The Go Authors
See source code for license details.

Support for streaming Protocol Buffer messages for the Go language (golang).
https://github.com/matttproud/golang_protobuf_extensions
Copyright 2013 Matt T. Proud
Licensed under the Apache License, Version 2.0

1.42 pgv-java-stubs 0.6.1

1.42.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,

publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution

notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing

the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.43 apache-commons-collections 3.2.2

1.43.1 Available under license :

Apache Commons Collections
Copyright 2001-2015 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

Apache License

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions

to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices

stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- 5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
- 6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
- 7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.44 netty-codec-redis 4.1.72.Final

1.44.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2018 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/InlineCommandRedisMessage.java
```

No license file was found, but licenses were detected in source scan.

```
<!--
 ~ Copyright 2016 The Netty Project
 ~
 ~ The Netty Project licenses this file to you under the Apache License,
 ~ version 2.0 (the "License"); you may not use this file except in compliance
 ~ with the License. You may obtain a copy of the License at:
 ~
 ~ https://www.apache.org/licenses/LICENSE-2.0
 ~
 ~ Unless required by applicable law or agreed to in writing, software
 ~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 ~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
```

~ License for the specific language governing permissions and limitations

~ under the License.

-->

Found in path(s):

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/META-INF/maven/io.netty/netty-codec-redis/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License, version 2.0 (the

* "License"); you may not use this file except in compliance with the License. You may obtain a

* copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/package-info.java

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/RedisMessagePool.java

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/ArrayRedisMessage.java

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/RedisCodecUtil.java

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/AbstractStringRedisMessage.java

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/LastBulkStringRedisContent.java

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/RedisMessage.java

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/ErrorRedisMessage.java

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/BulkStringHeaderRedisMessage.java

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/FixedRedisMessagePool.java

* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-jar/io/netty/handler/codec/redis/RedisDecoder.java

```
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/RedisCodecException.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/RedisBulkStringAggregator.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/RedisConstants.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/DefaultBulkStringRedisContent.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/RedisArrayAggregator.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/SimpleStringRedisMessage.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/RedisEncoder.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/BulkStringRedisContent.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/RedisMessageType.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/ArrayHeaderRedisMessage.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/IntegerRedisMessage.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/DefaultLastBulkStringRedisContent.java
* /opt/cola/permits/1411866752_1662683824.9374516/0/netty-codec-redis-4-1-72-final-sources-1-
jar/io/netty/handler/codec/redis/FullBulkStringRedisMessage.java
```

1.45 netty-project 4.1.68.Final

1.45.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<!--
~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->
```

Found in path(s):

* /opt/cola/permits/1331306087_1662690272.2162187/0/netty-transport-native-epoll-4-1-68-final-jar/META-INF/maven/io.netty/netty-transport-native-epoll/pom.xml

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Implementation-Title: Netty/Transport/Native/Epoll

Bundle-Description: Netty is an asynchronous event-driven network application framework for rapid development of maintainable high performance protocol servers and clients.

Automatic-Module-Name: io.netty.transport.epoll

Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0>

Bundle-SymbolicName: io.netty.transport-native-epoll

Implementation-Version: 4.1.68.Final

Built-By: root

Bnd-LastModified: 1631193593222

Bundle-ManifestVersion: 2

Implementation-Vendor-Id: io.netty

Bundle-DocURL: <https://netty.io/>

Bundle-Vendor: The Netty Project

Import-Package: io.netty.buffer;version="[4.1,5)",io.netty.channel,io.netty.channel.socket;version="[4.1,5)",io.netty.channel.unix;version="[4.1,5)",io.netty.util;version="[4.1,5)",io.netty.util.collection;version="[4.1,5)",io.netty.util.concurrent;version="[4.1,5)",io.netty.util.internal;version="[4.1,5)",io.netty.util.internal.logging;version="[4.1,5)",sun.nio.ch;resolution:=optional,org.eclipse.jetty.npn;version="[1,2)";resolution:=optional,org.eclipse.jetty.alpn;version="[1,2)";resolution:=optional

Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.6))"

Tool: Bnd-2.4.1.201501161923

Implementation-Vendor: The Netty Project

Export-Package: io.netty.channel.epoll;uses:="io.netty.buffer,io.netty.channel,io.netty.channel.socket,io.netty.channel.unix,io.netty.util,io.netty.util.concurrent";version="4.1.68"

Bundle-Name: Netty/Transport/Native/Epoll

Bundle-Version: 4.1.68.Final

Created-By: Apache Maven Bundle Plugin

Build-Jdk: 1.8.0_292

Implementation-URL: <https://netty.io/netty-transport-native-epoll/>

Found in path(s):

* /opt/cola/permits/1331306087_1662690272.2162187/0/netty-transport-native-epoll-4-1-68-final-jar/META-INF/MANIFEST.MF

1.46 burntsushi-toml v0.3.1

1.46.1 Available under license :

The MIT License (MIT)

Copyright (c) 2013 TOML authors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.47 jackson-core 2.12.5

1.47.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.

It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0 To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.48 protobuf v1.27.1

1.48.1 Available under license :

Copyright (c) 2018 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- # This source code was written by the Go contributors.
- # The master list of contributors is in the main Go distribution,
- # visible at <https://tip.golang.org/CONTRIBUTORS>.

1.49 aws-java-sdk-::third-party-::jackson-dataformat-cbor 2.17.52

1.49.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the

editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the

Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the

same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (<https://www.apache.org/>).

1.50 aws-java-sdk-for-amazon-dynamodb

1.12.26

1.50.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the

editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the

Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the

same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.51 jackson-annotations jackson- annotations-2.12.3

1.51.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.52 aws-java-sdk-::-services-::-amazon-cloudwatch 2.17.52

1.52.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to

communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of

the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (<https://www.apache.org/>).

1.53 aws-java-sdk-::-core-::-protocols-::-aws-cbor-protocol 2.17.52

1.53.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and

limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (<https://www.apache.org/>).

1.54 zstd-jni 1.4.9-1

1.54.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Automatic-Module-Name: com.github.luben.zstd_jni

Bnd-LastModified: 1615191527618

Bundle-Description: JNI bindings for Zstd native library that provides fast and high compression lossless algorithm for Java and all JVM languages.

Bundle-License: <https://opensource.org/licenses/BSD-2-Clause>;description=BSD 2-Clause License

Bundle-ManifestVersion: 2

Bundle-Name: zstd-jni

Bundle-NativeCode: aix/ppc64/libzstd-jni.so;osname=AIX;processor=ppc64
, darwin/x86_64/libzstd-jni.dylib;osname=MacOS;osname=MacOSX;processo
r=x86_64, darwin/aarch64/libzstd-jni.dylib;osname=MacOS;osname=MacOSX
;processor=aarch64, freebsd/amd64/libzstd-jni.so;osname=FreeBSD;proce
ssor=amd64, freebsd/i386/libzstd-jni.so;osname=FreeBSD;processor=i386
, linux/aarch64/libzstd-jni.so;osname=Linux;processor=aarch64, linux/
amd64/libzstd-jni.so;osname=Linux;processor=amd64, linux/arm/libzstd-
jni.so;osname=Linux;processor=arm, linux/i386/libzstd-jni.so;osname=L
inux;processor=i386, linux/mips64/libzstd-jni.so;osname=Linux;process
or=mips64, linux/ppc64/libzstd-jni.so;osname=Linux;processor=ppc64, l
inux/ppc64le/libzstd-jni.so;osname=Linux;processor=ppc64le, linux/s39
0x/libzstd-jni.so;osname=Linux;processor=s390x, win/amd64/libzstd-jni
.dll;osname=Win32;processor=amd64, win/x86/libzstd-jni.dll;osname=Win
32;processor=x86
Bundle-SymbolicName: com.github.luben.zstd-jni
Bundle-Vendor: com.github.luben
Bundle-Version: 1.4.9.1
Created-By: 1.8.0_275 (Debian)
Export-Package: com.github.luben.zstd;version="1.4.9.1",com.github.lub
en.zstd.util;version="1.4.9.1"
Implementation-Title: zstd-jni
Implementation-Vendor: com.github.luben
Implementation-Vendor-Id: com.github.luben
Implementation-Version: 1.4.9-1
Import-Package: org.osgi.framework;resolution:=optional
Private-Package: linux.amd64,linux.i386,linux.aarch64,linux.arm,linux.
ppc64,linux.ppc64le,linux.mips64,linux.s390x,aix.ppc64,darwin.x86_64,
darwin.aarch64,win.amd64,win.x86,freebsd.amd64,freebsd.i386
Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.8))"
Specification-Title: zstd-jni
Specification-Vendor: com.github.luben
Specification-Version: 1.4.9-1
Tool: Bnd-4.0.0.201805111645

Found in path(s):

* /opt/cola/permits/1183892379_1627494642.79/0/zstd-jni-1-4-9-1-1-jar/META-INF/MANIFEST.MF

1.55 aws-java-sdk-::-services-::-amazon-dynamodb 2.17.52

1.55.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:

Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (<https://www.apache.org/>).

1.56 apache-log4j-api 2.17.1

1.56.1 Available under license :

Apache Log4j API
Copyright 1999-1969 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (<http://www.apache.org/>).

Apache License

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions

to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices

stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- 5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
- 6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
- 7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.57 junit 4.13.1

1.57.1 Available under license :

JUnit

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

- a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
- b) in the case of each subsequent Contributor:

- i) changes to the Program, and

- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

- a) it complies with the terms and conditions of this Agreement; and
- b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor

to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights

granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

1.58 thanhpk/randstr v1.0.4

1.58.1 Available under license :

The MIT License

Copyright (c) 2010-2018 Google, Inc. <http://angularjs.org>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is

furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.59 aws-java-sdk-auth 2.17.52

1.59.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.60 fsnotify-fsnotify v1.4.9

1.60.1 Available under license :

Copyright (c) 2012 The Go Authors. All rights reserved.

Copyright (c) 2012-2019 fsnotify Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.61 checker-qual 3.5.0

1.61.1 Available under license :

Checker Framework qualifiers

Copyright 2004-present by the Checker Framework developers

MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.62 protocol-buffer-java-util-package 3.15.6

1.62.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Automatic-Module-Name: com.google.protobuf.util

Bnd-LastModified: 1615494724341

Build-Jdk: 1.8.0_181-google-v7

Built-By: acozzette

Bundle-Description: Utilities for Protocol Buffers

Bundle-DocURL: <https://developers.google.com/protocol-buffers/>

Bundle-License: <https://opensource.org/licenses/BSD-3-Clause>

Bundle-ManifestVersion: 2

Bundle-Name: Protocol Buffers [Util]

Bundle-SymbolicName: com.google.protobuf.util

Bundle-Version: 3.15.6

Created-By: Apache Maven Bundle Plugin

Export-Package: com.google.protobuf.util;version="3.15.6";uses:="com.google.protobuf"

Import-Package: com.google.common.base;version="[30.0,31)",com.google.common.io;version="[30.0,31)",com.google.common.math;version="[30.0,31)",com.google.common.primitives;version="[30.0,31)",com.google.gson;versio

n="[2.8,3)",com.google.gson.stream;version="[2.8,3)",com.google.protobuf;version="[3.15,4)"
Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.7))"
Tool: Bnd-3.0.0.201509101326

Found in path(s):

* /opt/cola/permits/1411866746_1662690446.511798/0/protobuf-java-util-3-15-6-jar/META-INF/MANIFEST.MF

1.63 io-grpc-grpc-protobuf 1.39.0

1.63.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017 The gRPC Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-jar/io/grpc/protobuf/package-info.java
* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-jar/io/grpc/protobuf/StatusProto.java
* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-jar/io/grpc/protobuf/ProtoMethodDescriptorSupplier.java
* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-jar/io/grpc/protobuf/ProtoServiceDescriptorSupplier.java

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2014 The gRPC Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
*/
```

```
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-
jar/io/grpc/protobuf/ProtoUtils.java
```

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2016 The gRPC Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-
jar/io/grpc/protobuf/ProtoFileDescriptorSupplier.java
```

1.64 aws-event-stream 1.0.1

1.64.1 Available under license :

MIT License

Copyright (c) 2018 Lifion, by ADP

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.65 jackson-databind 2.12.5

1.65.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.66 kotlin-scripting-common 1.3.50

1.66.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2010-2018 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1137595702_1614104283.49/0/kotlin-scripting-common-1-3-50-sources-3-
jar/kotlin/script/experimental/host/BasicScriptingHost.kt
```

1.67 aws-glue-schema-registry-common 1.1.5

1.67.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<!--
/*
 * Copyright 2020 Amazon.com, Inc. or its affiliates.
 * Licensed under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
-->
```

Found in path(s):

```
* /opt/cola/permits/1411866810_1662685628.3642151/0/schema-registry-common-1-1-5-jar/META-
```


1.68 netty-reactive-streams-implementation

2.0.5

1.68.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bundle-Description: Reactive streams implementation for Netty.
Automatic-Module-Name: com.typesafe.netty.core
Bundle-License: <http://www.apache.org/licenses/LICENSE-2.0.txt>
Bundle-SymbolicName: com.typesafe.netty.reactive-streams
Built-By: marcospereira
Bnd-LastModified: 1602622953847
Bundle-ManifestVersion: 2
Bundle-DocURL: <http://typesafe.com/>
Bundle-Vendor: Typesafe
Import-Package: io.netty.channel;version="[4.1,5)",io.netty.util;versi
on="[4.1,5)",io.netty.util.concurrent;version="[4.1,5)",io.netty.util
.internal;version="[4.1,5)",org.reactivestreams;version="[1.0,2)"
Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.7))"
Tool: Bnd-3.5.0.201709291849
Export-Package: com.typesafe.netty;uses:="io.netty.channel,io.netty.ut
il.concurrent,org.reactivestreams";version="2.0.5"
Bundle-Name: Netty Reactive Streams Implementation
Bundle-Version: 2.0.5
Created-By: Apache Maven Bundle Plugin
Build-Jdk: 1.8.0_181

Found in path(s):

* /opt/cola/permits/1128619582_1649176967.68/0/netty-reactive-streams-2-0-5-jar/META-INF/MANIFEST.MF

1.69 junit-jupiter-junit-jupiter-params 5.8.2

1.69.1 Available under license :

Apache License

=====

Version 2.0, January 2004

<<<https://www.apache.org/licenses/>>>>

Terms and Conditions for use, reproduction, and distribution

1. Definitions

License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
- * **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along

with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or

out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

- * **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
- * **b)** in the case of each subsequent Contributor:
 - * **i)** changes to the Program, and
 - * **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

* ****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the

Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)*** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

* **i)*** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* **ii)*** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

* **iii)*** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

* **iv)*** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)*** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* **b)*** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay

any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was

received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.70 perfmark-perfmark-api 0.23.0

1.70.1 Available under license :

Copyright 2019 Google LLC

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and

limitations under the License.

This product contains a modified portion of 'Catapult', an open source Trace Event viewer for Chrome, Linux, and Android applications, which can be obtained at:

* LICENSE:

* [traceviewer/src/main/resources/io/perfmark/traceviewer/third_party/catapult/LICENSE](https://github.com/catapult-project/catapult/blob/master/traceviewer/src/main/resources/io/perfmark/traceviewer/third_party/catapult/LICENSE) (New BSD License)

* HOMEPAGE:

* <https://github.com/catapult-project/catapult>

This product contains a modified portion of 'Polymer', a library for Web Components, which can be obtained at:

* LICENSE:

* [traceviewer/src/main/resources/io/perfmark/traceviewer/third_party/polymer/LICENSE](https://github.com/Polymer/polymer/blob/master/traceviewer/src/main/resources/io/perfmark/traceviewer/third_party/polymer/LICENSE) (New BSD License)

* HOMEPAGE:

* <https://github.com/Polymer/polymer>

Copyright 2015 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of catapult nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

// Copyright (c) 2012 The Polymer Authors. All rights reserved.

//

// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:

//

// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.

// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.

// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.

//

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*

* Copyright 2019 Google LLC

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

1.71 apache-commons-validator 1.7

1.71.1 Available under license :

Apache Commons Validator
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work

(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses

granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"

replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.72 mockito v3.11.2

1.72.1 Available under license :

The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.73 j2objc-annotations 1.3

1.73.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-
jar/com/google/j2objc/annotations/RetainedWith.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-
jar/com/google/j2objc/annotations/Property.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-
jar/com/google/j2objc/annotations/LoopTranslation.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-
jar/com/google/j2objc/annotations/ObjectiveCName.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-
jar/com/google/j2objc/annotations/ReflectionSupport.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-
jar/com/google/j2objc/annotations/RetainedLocalRef.java
* /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-
jar/com/google/j2objc/annotations/J2ObjCIncompatible.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2012 Google Inc. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
```

- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/Weak.java
- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/AutoreleasePool.java
- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/WeakOuter.java

1.74 lumberjack v2.0.0

1.74.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Nate Finch

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.75 aws-java-sdk-core-protocols-protocol-core 2.17.52

1.75.1 Available under license :

Apache License
Version 2.0, January 2004

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally

submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or

implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (<https://www.apache.org/>).

1.76 kotlin-script-runtime 1.3.50

1.76.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2010-2017 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1168630308_1622041949.67/0/kotlin-script-runtime-1-3-50-sources-4-jar/kotlin/script/extensions/samWithReceiver.kt
* /opt/cola/permits/1168630308_1622041949.67/0/kotlin-script-runtime-1-3-50-sources-4-jar/kotlin/script/dependencies/resolvers_deprecated.kt
* /opt/cola/permits/1168630308_1622041949.67/0/kotlin-script-runtime-1-3-50-sources-4-jar/kotlin/script/dependencies/dependencies_deprecated.kt
* /opt/cola/permits/1168630308_1622041949.67/0/kotlin-script-runtime-1-3-50-sources-4-jar/kotlin/script/experimental/dependencies/resolvers.kt
* /opt/cola/permits/1168630308_1622041949.67/0/kotlin-script-runtime-1-3-50-sources-4-jar/kotlin/script/templates/annotations.kt
* /opt/cola/permits/1168630308_1622041949.67/0/kotlin-script-runtime-1-3-50-sources-4-jar/kotlin/script/experimental/dependencies/dependencies.kt
* /opt/cola/permits/1168630308_1622041949.67/0/kotlin-script-runtime-1-3-50-sources-4-jar/kotlin/script/experimental/dependencies/AsyncDependenciesResolver.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2016 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1168630308_1622041949.67/0/kotlin-script-runtime-1-3-50-sources-4-jar/kotlin/script/templates/standard/templates.kt

1.77 aws-java-sdk-annotations 2.17.52

1.77.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade

names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier

identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and

limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed

with the accompanying software if such terms are included in the directory containing the accompanying software.

Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by

Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:

Apache Commons Lang

Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<https://www.apache.org/>).

1.78 apache-log4j 2.17.1

1.78.1 Available under license :

Apache Log4j Core
Copyright 1999-2012 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or

Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work

or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work

by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.79 aws-java-sdk-secretsmanager 1.11.409

1.79.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance
 * with
 * the License. A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
 * permissions
 * and limitations under the License.
 */
/**
 * <p>
 * Creates a new secret. A secret in Secrets Manager consists of both the protected secret data and the important
 * information needed to manage the secret.
```

* </p>

* <p>

* Secrets Manager stores the encrypted secret data in one of a collection of "versions" associated with the secret.

* Each version contains a copy of the encrypted secret data. Each version is associated with one or more

* "staging labels" that identify where the version is in the rotation cycle. The

* <code>SecretVersionsToStages</code> field of the secret contains the mapping of staging labels to the active

* versions of the secret. Versions without a staging label are considered deprecated and are not included in the

* list.

* </p>

* <p>

* You provide the secret data to be encrypted by putting text in either the <code>SecretString</code> parameter

or

* binary data in the <code>SecretBinary</code> parameter, but not both. If you include

<code>SecretString</code> or

* <code>SecretBinary</code> then Secrets Manager also creates an initial secret version and automatically

attaches

* the staging label <code>AWSCURRENT</code> to the new version.

* </p>

* <note>

*

*

* <p>

* If you call an operation that needs to encrypt or decrypt the <code>SecretString</code> or

* <code>SecretBinary</code> for a secret in the same account as the calling user and that secret doesn't specify a

* AWS KMS encryption key, Secrets Manager uses the account's default AWS managed customer master key

(CMK) with the

* alias <code>aws/secretsmanager</code>. If this key doesn't already exist in your account then Secrets Manager

* creates it for you automatically. All users in the same AWS account automatically have access to use the default

* CMK. Note that if an Secrets Manager API call results in AWS having to create the account's AWS-managed

CMK, it

* can result in a one-time significant delay in returning the result.

* </p>

*

*

* <p>

* If the secret is in a different AWS account from the credentials calling an API that requires encryption or

* decryption of the secret value then you must create and use a custom AWS KMS CMK because you can't access

the

* default CMK for the account using credentials from a different AWS account. Store the ARN of the CMK in the

* secret when you create the secret or when you update it by including it in the <code>KMSKeyId</code>. If you

call

* an API that must encrypt or decrypt <code>SecretString</code> or <code>SecretBinary</code> using

credentials from

* a different account then the AWS KMS key policy must grant cross-account access to that other account's user

or

* role for both the kms:GenerateDataKey and kms:Decrypt operations.

* </p>

*

*

* </note>

* <p>

* </p>

* <p>

* Minimum permissions

* </p>

* <p>

* To run this command, you must have the following permissions:

* </p>

*

*

* <p>

* secretsmanager:CreateSecret

* </p>

*

*

* <p>

* kms:GenerateDataKey - needed only if you use a customer-managed AWS KMS key to encrypt the secret. You do not

do not

* need this permission to use the account's default AWS managed CMK for Secrets Manager.

* </p>

*

*

* <p>

* kms:Decrypt - needed only if you use a customer-managed AWS KMS key to encrypt the secret. You do not

need this

* permission to use the account's default AWS managed CMK for Secrets Manager.

* </p>

*

*

* <p>

* Related operations

* </p>

*

*

* <p>

* To delete a secret, use <a>DeleteSecret.

* </p>

*

*

* <p>

* To modify an existing secret, use <a>UpdateSecret.

* </p>

*

*

* <p>

* To create a new version of a secret, use <a>PutSecretValue.


```

* </p>
* </li>
* <li>
* <p>
* To retrieve the encrypted secure string and secure binary values, use <a>GetSecretValue</a>.
* </p>
* </li>
* <li>
* <p>
* To retrieve all other details for a secret, use <a>DescribeSecret</a>. This does not include the encrypted secure
* string and secure binary values.
* </p>
* </li>
* <li>
* <p>
* To retrieve the list of secret versions associated with the current secret, use <a>DescribeSecret</a> and examine
* the <code>SecretVersionsToStages</code> response value.
* </p>
* </li>
* </ul>
*
* @param createSecretRequest
* @return A Java Future containing the result of the CreateSecret operation returned by the service.
* @sample AWSSecretsManagerAsync.CreateSecret
* @see <a href="http://docs.aws.amazon.com/goto/WebAPI/secretsmanager-2017-10-17/CreateSecret"
target="_top">AWS
*   API Documentation</a>
*/
/**
* <p>
* Stores a new encrypted secret value in the specified secret. To do this, the operation creates a new version and
* attaches it to the secret. The version can contain a new <code>SecretString</code> value or a new
* <code>SecretBinary</code> value. You can also specify the staging labels that are initially attached to the new
* version.
* </p>
* <note>
* <p>
* The Secrets Manager console uses only the <code>SecretString</code> field. To add binary data to a secret
with
* the <code>SecretBinary</code> field you must use the AWS CLI or one of the AWS SDKs.
* </p>
* </note>
* <ul>
* <li>
* <p>
* If this operation creates the first version for the secret then Secrets Manager automatically attaches the
* staging label <code>AWSCURRENT</code> to the new version.
* </p>

```

- *
- *
- * <p>
- * If another version of this secret already exists, then this operation does not automatically move any staging labels other than those that you explicitly specify in the `VersionStages` parameter.
- * </p>
- *
- *
- * <p>
- * If this operation moves the staging label `AWSCURRENT` from another version to this version (because you included it in the `StagingLabels` parameter) then Secrets Manager also automatically moves the staging label `AWSPREVIOUS` to the version that `AWSCURRENT` was removed from.
- * </p>
- *
- *
- * <p>
- * This operation is idempotent. If a version with a `VersionId` with the same value as the `ClientRequestToken` parameter already exists and you specify the same secret data, the operation succeeds but does nothing. However, if the secret data is different, then the operation fails because you cannot modify an existing version; you can only create new ones.
- * </p>
- *
- *
- * <note>
- *
- *
- * <p>
- * If you call an operation that needs to encrypt or decrypt the `SecretString` or `SecretBinary` for a secret in the same account as the calling user and that secret doesn't specify a AWS KMS encryption key, Secrets Manager uses the account's default AWS managed customer master key (CMK) with the alias `aws/secretsmanager`. If this key doesn't already exist in your account then Secrets Manager creates it for you automatically. All users in the same AWS account automatically have access to use the default CMK. Note that if an Secrets Manager API call results in AWS having to create the account's AWS-managed CMK, it can result in a one-time significant delay in returning the result.
- * </p>
- *
- *
- * <p>
- * If the secret is in a different AWS account from the credentials calling an API that requires encryption or decryption of the secret value then you must create and use a custom AWS KMS CMK because you can't access the default CMK for the account using credentials from a different AWS account. Store the ARN of the CMK in the

* secret when you create the secret or when you update it by including it in the `KMSKeyId`. If you call

* an API that must encrypt or decrypt `SecretString` or `SecretBinary` using credentials from

* a different account then the AWS KMS key policy must grant cross-account access to that other account's user or

* role for both the `kms:GenerateDataKey` and `kms:Decrypt` operations.

* `</p>`

* ``

* ``

* `</note>`

* `<p>`

* **Minimum permissions**

* `</p>`

* `<p>`

* To run this command, you must have the following permissions:

* `</p>`

* ``

* ``

* `<p>`

* `secretsmanager:PutSecretValue`

* `</p>`

* ``

* ``

* `<p>`

* `kms:GenerateDataKey` - needed only if you use a customer-managed AWS KMS key to encrypt the secret. You do not

* need this permission to use the account's default AWS managed CMK for Secrets Manager.

* `</p>`

* ``

* ``

* `<p>`

* **Related operations**

* `</p>`

* ``

* ``

* `<p>`

* To retrieve the encrypted value you store in the version of a secret, use `GetSecretValue`.

* `</p>`

* ``

* ``

* `<p>`

* To create a secret, use `CreateSecret`.

* `</p>`

* ``

* ``

* `<p>`

* To get the details for a secret, use `DescribeSecret`.

```

* </p>
* </li>
* <li>
* <p>
* To list the versions attached to a secret, use <a>ListSecretVersionIds</a>.
* </p>
* </li>
* </ul>
*
* @param putSecretValueRequest
* @return A Java Future containing the result of the PutSecretValue operation returned by the service.
* @sample AWSSecretsManagerAsync.PutSecretValue
* @see <a href="http://docs.aws.amazon.com/goto/WebAPI/secretsmanager-2017-10-17/PutSecretValue"
target="_top">AWS
*   API Documentation</a>
*/
/**
* <p>
* Modifies many of the details of the specified secret. If you include a <code>ClientRequestToken</code> and
* <i>either</i> <code>SecretString</code> or <code>SecretBinary</code> then it also creates a new version
attached
* to the secret.
* </p>
* <p>
* To modify the rotation configuration of a secret, use <a>RotateSecret</a> instead.
* </p>
* <note>
* <p>
* The Secrets Manager console uses only the <code>SecretString</code> parameter and therefore limits you to
* encrypting and storing only a text string. To encrypt and store binary data as part of the version of a secret,
* you must use either the AWS CLI or one of the AWS SDKs.
* </p>
* </note>
* <ul>
* <li>
* <p>
* If a version with a <code>VersionId</code> with the same value as the <code>ClientRequestToken</code>
parameter
* already exists, the operation results in an error. You cannot modify an existing version, you can only create a
* new version.
* </p>
* </li>
* <li>
* <p>
* If you include <code>SecretString</code> or <code>SecretBinary</code> to create a new secret version,
Secrets
* Manager automatically attaches the staging label <code>AWSCURRENT</code> to the new version.
* </p>

```

- *
- *
- * <note>
- *
- *
- * <p>
- * If you call an operation that needs to encrypt or decrypt the <code>SecretString</code> or
- * <code>SecretBinary</code> for a secret in the same account as the calling user and that secret doesn't specify a
- * AWS KMS encryption key, Secrets Manager uses the account's default AWS managed customer master key
- (CMK) with the
- * alias <code>aws/secretsmanager</code>. If this key doesn't already exist in your account then Secrets Manager
- * creates it for you automatically. All users in the same AWS account automatically have access to use the default
- * CMK. Note that if an Secrets Manager API call results in AWS having to create the account's AWS-managed
- CMK, it
- * can result in a one-time significant delay in returning the result.
- * </p>
- *
- *
- * <p>
- * If the secret is in a different AWS account from the credentials calling an API that requires encryption or
- * decryption of the secret value then you must create and use a custom AWS KMS CMK because you can't access
- the
- * default CMK for the account using credentials from a different AWS account. Store the ARN of the CMK in the
- * secret when you create the secret or when you update it by including it in the <code>KMSKeyId</code>. If you
- call
- * an API that must encrypt or decrypt <code>SecretString</code> or <code>SecretBinary</code> using
- credentials from
- * a different account then the AWS KMS key policy must grant cross-account access to that other account's user
- or
- * role for both the kms:GenerateDataKey and kms:Decrypt operations.
- * </p>
- *
- *
- * </note>
- * <p>
- * Minimum permissions
- * </p>
- * <p>
- * To run this command, you must have the following permissions:
- * </p>
- *
- *
- * <p>
- * secretsmanager:UpdateSecret
- * </p>
- *
- *
- * <p>

```

* kms:GenerateDataKey - needed only if you use a custom AWS KMS key to encrypt the secret. You do not need
this
* permission to use the account's AWS managed CMK for Secrets Manager.
* </p>
* </li>
* <li>
* <p>
* kms:Decrypt - needed only if you use a custom AWS KMS key to encrypt the secret. You do not need this
permission
* to use the account's AWS managed CMK for Secrets Manager.
* </p>
* </li>
* </ul>
* <p>
* <b>Related operations</b>
* </p>
* <ul>
* <li>
* <p>
* To create a new secret, use <a>CreateSecret</a>.
* </p>
* </li>
* <li>
* <p>
* To add only a new version to an existing secret, use <a>PutSecretValue</a>.
* </p>
* </li>
* <li>
* <p>
* To get the details for a secret, use <a>DescribeSecret</a>.
* </p>
* </li>
* <li>
* <p>
* To list the versions contained in a secret, use <a>ListSecretVersionIds</a>.
* </p>
* </li>
* </ul>
*
* @param updateSecretRequest
* @return A Java Future containing the result of the UpdateSecret operation returned by the service.
* @sample AWSSecretsManagerAsync.UpdateSecret
* @see <a href="http://docs.aws.amazon.com/goto/WebAPI/secretsmanager-2017-10-17/UpdateSecret"
target="_top">AWS
*   API Documentation</a>
*/

```

Found in path(s):

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/AWSSecretsManagerAsync.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.

*

* Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with

* the License. A copy of the License is located at

*

* <http://aws.amazon.com/apache2.0>

*

* or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR

* CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions

* and limitations under the License.

*/

/**

* <p>

* Creates a new secret. A secret in Secrets Manager consists of both the protected secret data and the important information needed to manage the secret.

* </p>

* <p>

* Secrets Manager stores the encrypted secret data in one of a collection of "versions" associated with the secret.

* Each version contains a copy of the encrypted secret data. Each version is associated with one or more

* "staging labels" that identify where the version is in the rotation cycle. The

* `SecretVersionsToStages` field of the secret contains the mapping of staging labels to the active

* versions of the secret. Versions without a staging label are considered deprecated and are not included in the

* list.

* </p>

* <p>

* You provide the secret data to be encrypted by putting text in either the `SecretString` parameter or

* binary data in the `SecretBinary` parameter, but not both. If you include

`SecretString` or

* `SecretBinary` then Secrets Manager also creates an initial secret version and automatically attaches

* the staging label `AWSCURRENT` to the new version.

* </p>

* <note>

*

*

* <p>

* If you call an operation that needs to encrypt or decrypt the `SecretString` or

* `SecretBinary` for a secret in the same account as the calling user and that secret doesn't specify a

* AWS KMS encryption key, Secrets Manager uses the account's default AWS managed customer master key

(CMK) with the

- * `alias aws/secretsmanager`. If this key doesn't already exist in your account then Secrets Manager
- * creates it for you automatically. All users in the same AWS account automatically have access to use the default
- * CMK. Note that if an Secrets Manager API call results in AWS having to create the account's AWS-managed

CMK, it

- * can result in a one-time significant delay in returning the result.

* `</p>`

* ``

* ``

* `<p>`

- * If the secret is in a different AWS account from the credentials calling an API that requires encryption or

* decryption of the secret value then you must create and use a custom AWS KMS CMK because you can't access the

* default CMK for the account using credentials from a different AWS account. Store the ARN of the CMK in the

secret when you create the secret or when you update it by including it in the `KMSKeyId`. If you

call an API that must encrypt or decrypt `SecretString` or `SecretBinary` using credentials from

a different account then the AWS KMS key policy must grant cross-account access to that other account's user

or

- * role for both the `kms:GenerateDataKey` and `kms:Decrypt` operations.

* `</p>`

* ``

* ``

* `</note>`

* `<p>`

* `</p>`

* `<p>`

* **Minimum permissions**

* `</p>`

* `<p>`

* To run this command, you must have the following permissions:

* `</p>`

* ``

* ``

* `<p>`

* `secretsmanager:CreateSecret`

* `</p>`

* ``

* ``

* `<p>`

* `kms:GenerateDataKey` - needed only if you use a customer-managed AWS KMS key to encrypt the secret. You do not

* need this permission to use the account's default AWS managed CMK for Secrets Manager.

* `</p>`

* ``

* ``

* `<p>`

* kms:Decrypt - needed only if you use a customer-managed AWS KMS key to encrypt the secret. You do not need this

* permission to use the account's default AWS managed CMK for Secrets Manager.

* </p>

*

*

* <p>

* Related operations

* </p>

*

*

* <p>

* To delete a secret, use <a>DeleteSecret.

* </p>

*

*

* <p>

* To modify an existing secret, use <a>UpdateSecret.

* </p>

*

*

* <p>

* To create a new version of a secret, use <a>PutSecretValue.

* </p>

*

*

* <p>

* To retrieve the encrypted secure string and secure binary values, use <a>GetSecretValue.

* </p>

*

*

* <p>

* To retrieve all other details for a secret, use <a>DescribeSecret. This does not include the encrypted secure string and secure binary values.

* </p>

*

*

* <p>

* To retrieve the list of secret versions associated with the current secret, use <a>DescribeSecret and examine the <code>SecretVersionsToStages</code> response value.

* </p>

*

*

*

* @param createSecretRequest

* @return Result of the CreateSecret operation returned by the service.

* @throws InvalidParameterException

* You provided an invalid value for a parameter.

```

* @throws InvalidRequestException
*     You provided a parameter value that is not valid for the current state of the resource.</p>
*     <p>
*     Possible causes:
*     </p>
*     <ul>
*     <li>
*     <p>
*     You tried to perform the operation on a secret that's currently marked deleted.
*     </p>
*     </li>
*     <li>
*     <p>
*     You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and
*     you didn't include such an ARN as a parameter in this call.
*     </p>
*     </li>
* @throws LimitExceededException
*     The request failed because it would exceed one of the Secrets Manager internal limits.
* @throws EncryptionFailureException
*     Secrets Manager can't encrypt the protected secret text using the provided KMS key. Check that the
*     customer master key (CMK) is available, enabled, and not in an invalid state. For more information, see
*     <a href="http://docs.aws.amazon.com/kms/latest/developerguide/key-state.html">How Key State Affects
Use
*     of a Customer Master Key</a>.
* @throws ResourceExistsException
*     A resource with the ID you requested already exists.
* @throws ResourceNotFoundException
*     We can't find the resource that you asked for.
* @throws MalformedPolicyDocumentException
*     The policy document that you provided isn't valid.
* @throws InternalServiceErrorException
*     An error occurred on the server side.
* @throws PreconditionNotMetException
*     The request failed because you did not complete all the prerequisite steps.
* @sample AWSSecretsManager.CreateSecret
* @see <a href="http://docs.aws.amazon.com/goto/WebAPI/secretsmanager-2017-10-17/CreateSecret"
target="_top">AWS
*     API Documentation</a>
*/
/**
* <p>
* Stores a new encrypted secret value in the specified secret. To do this, the operation creates a new version and
* attaches it to the secret. The version can contain a new <code>SecretString</code> value or a new
* <code>SecretBinary</code> value. You can also specify the staging labels that are initially attached to the new
* version.
* </p>
* <note>

```

* <p>

* The Secrets Manager console uses only the <code>SecretString</code> field. To add binary data to a secret with

* the <code>SecretBinary</code> field you must use the AWS CLI or one of the AWS SDKs.

* </p>

* </note>

*

*

* <p>

* If this operation creates the first version for the secret then Secrets Manager automatically attaches the

* staging label <code>AWSCURRENT</code> to the new version.

* </p>

*

*

* <p>

* If another version of this secret already exists, then this operation does not automatically move any staging

* labels other than those that you explicitly specify in the <code>VersionStages</code> parameter.

* </p>

*

*

* <p>

* If this operation moves the staging label <code>AWSCURRENT</code> from another version to this version (because

* you included it in the <code>StagingLabels</code> parameter) then Secrets Manager also automatically moves the

* staging label <code>AWSPREVIOUS</code> to the version that <code>AWSCURRENT</code> was removed from.

* </p>

*

*

* <p>

* This operation is idempotent. If a version with a <code>VersionId</code> with the same value as the

* <code>ClientRequestToken</code> parameter already exists and you specify the same secret data, the operation

* succeeds but does nothing. However, if the secret data is different, then the operation fails because you cannot

* modify an existing version; you can only create new ones.

* </p>

*

*

* <note>

*

*

* <p>

* If you call an operation that needs to encrypt or decrypt the <code>SecretString</code> or

* <code>SecretBinary</code> for a secret in the same account as the calling user and that secret doesn't specify a

* AWS KMS encryption key, Secrets Manager uses the account's default AWS managed customer master key (CMK) with the

* alias <code>aws/secretsmanager</code>. If this key doesn't already exist in your account then Secrets Manager

- * creates it for you automatically. All users in the same AWS account automatically have access to use the default CMK. Note that if an Secrets Manager API call results in AWS having to create the account's AWS-managed CMK, it
 - * can result in a one-time significant delay in returning the result.
- *
- * -
- * -
- *
- * If the secret is in a different AWS account from the credentials calling an API that requires encryption or decryption of the secret value then you must create and use a custom AWS KMS CMK because you can't access the
 - * default CMK for the account using credentials from a different AWS account. Store the ARN of the CMK in the
 - * secret when you create the secret or when you update it by including it in the `KMSKeyId`. If you call
 - * an API that must encrypt or decrypt `SecretString` or `SecretBinary` using credentials from
 - * a different account then the AWS KMS key policy must grant cross-account access to that other account's user or
 - * role for both the `kms:GenerateDataKey` and `kms:Decrypt` operations.
- *
- * -
- * -
- * -
- *
- * **Minimum permissions**
- *
- *
- * To run this command, you must have the following permissions:
 - *
 - *
- * -
- *
- * `secretsmanager:PutSecretValue`
- *
- * -
- * -
- *
- * `kms:GenerateDataKey` - needed only if you use a customer-managed AWS KMS key to encrypt the secret. You do not
 - * need this permission to use the account's default AWS managed CMK for Secrets Manager.

- *
- * -
- * -
- *
- * **Related operations**
- *
- *
- * -

* <p>

* To retrieve the encrypted value you store in the version of a secret, use <a>GetSecretValue.

* </p>

*

*

* <p>

* To create a secret, use <a>CreateSecret.

* </p>

*

*

* <p>

* To get the details for a secret, use <a>DescribeSecret.

* </p>

*

*

* <p>

* To list the versions attached to a secret, use <a>ListSecretVersionIds.

* </p>

*

*

*

* @param putSecretValueRequest

* @return Result of the PutSecretValue operation returned by the service.

* @throws InvalidParameterException

* You provided an invalid value for a parameter.

* @throws InvalidRequestException

* You provided a parameter value that is not valid for the current state of the resource.</p>

* <p>

* Possible causes:

* </p>

*

*

* <p>

* You tried to perform the operation on a secret that's currently marked deleted.

* </p>

*

*

* <p>

* You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.

* </p>

*

* @throws LimitExceededException

* The request failed because it would exceed one of the Secrets Manager internal limits.

* @throws EncryptionFailureException

* Secrets Manager can't encrypt the protected secret text using the provided KMS key. Check that the customer master key (CMK) is available, enabled, and not in an invalid state. For more information, see How Key State Affects

Use

- * of a [Customer Master Key](#).
- * @throws ResourceExistsException
- * A resource with the ID you requested already exists.
- * @throws ResourceNotFoundException
- * We can't find the resource that you asked for.
- * @throws InternalServiceErrorException
- * An error occurred on the server side.
- * @sample AWSecretsManager.PutSecretValue
- * @see <http://docs.aws.amazon.com/goto/WebAPI/secretsmanager-2017-10-17/PutSecretValue>

target="_top">AWS

- * [API Documentation](#)
- */

/**

- * <p>
- * Modifies many of the details of the specified secret. If you include a `ClientRequestToken` and
- * *either* `SecretString` or `SecretBinary` then it also creates a new version

attached

- * to the secret.
- * </p>
- * <p>
- * To modify the rotation configuration of a secret, use [RotateSecret](#) instead.
- * </p>
- * <note>
- * <p>
- * The Secrets Manager console uses only the `SecretString` parameter and therefore limits you to
- * encrypting and storing only a text string. To encrypt and store binary data as part of the version of a secret,
- * you must use either the AWS CLI or one of the AWS SDKs.
- * </p>
- * </note>
- *
- *
- * <p>
- * If a version with a `VersionId` with the same value as the `ClientRequestToken`

parameter

- * already exists, the operation results in an error. You cannot modify an existing version, you can only create a
- * new version.
- * </p>
- *
- *
- * <p>
- * If you include `SecretString` or `SecretBinary` to create a new secret version,

Secrets

- * Manager automatically attaches the staging label `AWSCURRENT` to the new version.
- * </p>
- *
- *
- * <note>

- * ``
- * ``
- * `<p>`
- * If you call an operation that needs to encrypt or decrypt the `<code>SecretString</code>` or `<code>SecretBinary</code>` for a secret in the same account as the calling user and that secret doesn't specify a `<code>AWS KMS encryption key</code>`, Secrets Manager uses the account's default AWS managed customer master key (CMK) with the
 - * `<code>alias aws/secretsmanager</code>`. If this key doesn't already exist in your account then Secrets Manager creates it for you automatically. All users in the same AWS account automatically have access to use the default CMK. Note that if an Secrets Manager API call results in AWS having to create the account's AWS-managed CMK, it
 - * can result in a one-time significant delay in returning the result.
- * `</p>`
- * ``
- * ``
- * `<p>`
- * If the secret is in a different AWS account from the credentials calling an API that requires encryption or decryption of the secret value then you must create and use a custom AWS KMS CMK because you can't access the
 - * default CMK for the account using credentials from a different AWS account. Store the ARN of the CMK in the secret when you create the secret or when you update it by including it in the `<code>KMSKeyId</code>`. If you call
 - * an API that must encrypt or decrypt `<code>SecretString</code>` or `<code>SecretBinary</code>` using credentials from
 - * a different account then the AWS KMS key policy must grant cross-account access to that other account's user or
 - * role for both the `kms:GenerateDataKey` and `kms:Decrypt` operations.
- * `</p>`
- * ``
- * ``
- * `</note>`
- * `<p>`
- * **Minimum permissions**
- * `</p>`
- * `<p>`
- * To run this command, you must have the following permissions:
 - * `<p>`
 - * ``
 - * ``
 - * `<p>`
 - * `secretsmanager:UpdateSecret`
 - * `</p>`
 - * ``
 - * ``
 - * `<p>`
 - * `kms:GenerateDataKey` - needed only if you use a custom AWS KMS key to encrypt the secret. You do not need this
 - * permission to use the account's AWS managed CMK for Secrets Manager.

* </p>

*

*

* <p>

* kms:Decrypt - needed only if you use a custom AWS KMS key to encrypt the secret. You do not need this permission

* to use the account's AWS managed CMK for Secrets Manager.

* </p>

*

*

* <p>

* Related operations

* </p>

*

*

* <p>

* To create a new secret, use <a>CreateSecret.

* </p>

*

*

* <p>

* To add only a new version to an existing secret, use <a>PutSecretValue.

* </p>

*

*

* <p>

* To get the details for a secret, use <a>DescribeSecret.

* </p>

*

*

* <p>

* To list the versions contained in a secret, use <a>ListSecretVersionIds.

* </p>

*

*

*

* @param updateSecretRequest

* @return Result of the UpdateSecret operation returned by the service.

* @throws InvalidParameterException

* You provided an invalid value for a parameter.

* @throws InvalidRequestException

* You provided a parameter value that is not valid for the current state of the resource.</p>

* <p>

* Possible causes:

* </p>

*

*

* <p>


```

*   You tried to perform the operation on a secret that's currently marked deleted.
*   </p>
*   </li>
*   <li>
*   <p>
*   You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and
*   you didn't include such an ARN as a parameter in this call.
*   </p>
*   </li>
* @throws LimitExceededException
*   The request failed because it would exceed one of the Secrets Manager internal limits.
* @throws EncryptionFailureException
*   Secrets Manager can't encrypt the protected secret text using the provided KMS key. Check that the
*   customer master key (CMK) is available, enabled, and not in an invalid state. For more information, see
*   <a href="http://docs.aws.amazon.com/kms/latest/developerguide/key-state.html">How Key State Affects
Use
*   of a Customer Master Key</a>.
* @throws ResourceExistsException
*   A resource with the ID you requested already exists.
* @throws ResourceNotFoundException
*   We can't find the resource that you asked for.
* @throws MalformedPolicyDocumentException
*   The policy document that you provided isn't valid.
* @throws InternalServiceErrorException
*   An error occurred on the server side.
* @throws PreconditionNotMetException
*   The request failed because you did not complete all the prerequisite steps.
* @sample AWSSecretsManager.UpdateSecret
* @see <a href="http://docs.aws.amazon.com/goto/WebAPI/secretsmanager-2017-10-17/UpdateSecret"
target="_top">AWS
*   API Documentation</a>
*/

```

Found in path(s):

```

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/AWSSecretsManagerClient.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/AWSSecretsManager.java

```

No license file was found, but licenses were detected in source scan.

```

/*
* Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance
with
* the License. A copy of the License is located at
*
* http://aws.amazon.com/apache2.0

```

*

* or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR

* CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions

* and limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/PutSecretValueRequestMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/CancelRotateSecretRequestMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/UpdateSecretRequestProtocolMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/DescribeSecretRequest.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/GetRandomPasswordResult.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/CreateSecretRequestMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/PutResourcePolicyRequestProtocolMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/RotateSecretResultJsonUnmarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/UpdateSecretRequestMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/GetRandomPasswordRequestProtocolMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/ListSecretVersionIdsRequest.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/RotationRulesTypeMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/Tag.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/GetResourcePolicyResult.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/PutResourcePolicyRequestMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/GetResourcePolicyRequestMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/transform/DescribeSecretRequestMarshaller.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/CancelRotateSecretRequest.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-jar/com/amazonaws/services/secretsmanager/model/GetResourcePolicyRequest.java

* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-

jar/com/amazonaws/services/secretsmanager/model/transform/GetSecretValueRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/TagMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/GetSecretValueResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/UpdateSecretVersionStageRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/DeleteResourcePolicyResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/DescribeSecretResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/GetResourcePolicyRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/MalformedPolicyDocumentException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/ResourceExistsException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/RotateSecretRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/ListSecretsRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/AWSecretsManagerClientBuilder.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/DeleteSecretRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/ListSecretVersionIdsRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/UpdateSecretResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/DeleteResourcePolicyRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/GetSecretValueResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/UntagResourceRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/TagResourceResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/PutResourcePolicyResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/TagResourceRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/SecretVersionsListEntryMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/ListSecretVersionIdsResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-

jar/com/amazonaws/services/secretsmanager/model/transform/SecretVersionsListEntryJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/ListSecretVersionIdsResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/CreateSecretResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/InvalidNextTokenException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/RotationRulesTypeJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/ListSecretsRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/ListSecretsRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/TagResourceRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/UntagResourceResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/EncryptionFailureException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/PutResourcePolicyResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/CreateSecretRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/DeleteResourcePolicyRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/AWSSecretsManagerException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/ListSecretsResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/GetRandomPasswordResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/UpdateSecretVersionStageResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/LimitExceededException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/AWSSecretsManagerAsyncClientBuilder.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/GetRandomPasswordRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/InvalidRequestException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/UpdateSecretVersionStageRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/DeleteResourcePolicyRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/DeleteSecretRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-

jar/com/amazonaws/services/secretsmanager/model/transform/TagJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/ListSecretVersionIdsRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/PutResourcePolicyRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/DeleteSecretResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/DeleteResourcePolicyResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/DeleteSecretRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/SecretListEntryJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/PutSecretValueRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/DescribeSecretRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/DeleteSecretResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/RestoreSecretRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/RotateSecretRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/TagResourceRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/PutSecretValueResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/package-info.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/UntagResourceResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/ResourceNotFoundException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/DescribeSecretResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/GetRandomPasswordRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/TagResourceResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/SecretVersionsListEntry.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/UpdateSecretRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/GetSecretValueRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/AbstractAWSSecretsManager.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-

jar/com/amazonaws/services/secretsmanager/model/transform/RotateSecretRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/CancelRotateSecretResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/InvalidParameterException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/RestoreSecretRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/PutSecretValueResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/ListSecretsResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/RestoreSecretResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/RotateSecretResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/SecretListEntryMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/CreateSecretRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/CancelRotateSecretResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/AbstractAWSSecretsManagerAsync.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/UntagResourceRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/CancelRotateSecretRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/DecryptionFailureException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/SecretListEntry.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/PreconditionNotMetException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/CreateSecretResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/AWSSecretsManagerAsyncClient.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/GetResourcePolicyResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/UpdateSecretVersionStageRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/UpdateSecretResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/PutSecretValueRequest.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/RestoreSecretRequestMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-

jar/com/amazonaws/services/secretsmanager/model/InternalServiceErrorException.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/UpdateSecretVersionStageResultJsonUnmarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/RotationRulesType.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/transform/UntagResourceRequestProtocolMarshaller.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/RestoreSecretResult.java
* /opt/cola/permits/1411866792_1662683683.0473826/0/aws-java-sdk-secretsmanager-1-11-409-sources-
jar/com/amazonaws/services/secretsmanager/model/GetSecretValueRequest.java

1.80 snappy-java 1.1.8.1

1.80.1 Available under license :

This product includes software developed by Google
Snappy: <http://code.google.com/p/snappy/> (New BSD License)

This product includes software developed by Apache
PureJavaCrc32C from apache-hadoop-common <http://hadoop.apache.org/>
(Apache 2.0 license)

This library contained statically linked libstdc++. This inclusion is allowed by
"GCC Runtime Library Exception"
<http://gcc.gnu.org/onlinedocs/libstdc++/manual/license.html>

== Contributors ==

- * Tatu Saloranta
 - * Providing benchmark suite
- * Alec Wysoker
 - * Performance and memory usage improvement

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.81 mac-os 4.1.72.Final

1.81.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<!--
~ Copyright 2021 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->
```

Found in path(s):

```
* /opt/cola/permits/1411866811_1662689138.833735/0/netty-resolver-dns-classes-macos-4-1-72-final-sources-1-
jar/META-INF/maven/io.netty/netty-resolver-dns-classes-macos/pom.xml
```

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1411866811_1662689138.833735/0/netty-resolver-dns-classes-macos-4-1-72-final-sources-1-
jar/io/netty/resolver/dns/macos/DnsResolver.java
* /opt/cola/permits/1411866811_1662689138.833735/0/netty-resolver-dns-classes-macos-4-1-72-final-sources-1-
jar/io/netty/resolver/dns/macos/package-info.java
* /opt/cola/permits/1411866811_1662689138.833735/0/netty-resolver-dns-classes-macos-4-1-72-final-sources-1-
jar/io/netty/resolver/dns/macos/MacOSDnsServerAddressStreamProvider.java
```

1.82 io-grpc-grpc-testing 1.39.0

1.82.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade

names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier

identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.83 jackson-dataformats-binary 2.12.3

1.83.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Bnd-LastModified: 1618252665347

Build-Jdk-Spec: 1.8

Bundle-Description: Support for reading and writing Concise Binary Object Representation([CBOR](https://www.rfc-editor.org/info/rfc7049)) encoded data using Jackson abstractions (streaming API, data binding, tree model)

Bundle-DocURL: <http://github.com/FasterXML/jackson-dataformats-binary>

Bundle-License: <http://www.apache.org/licenses/LICENSE-2.0.txt>

Bundle-ManifestVersion: 2

Bundle-Name: Jackson dataformat: CBOR

Bundle-SymbolicName: com.fasterxml.jackson.dataformat.jackson-dataformat-cbor

Bundle-Vendor: FasterXML

Bundle-Version: 2.12.3

Created-By: Apache Maven Bundle Plugin

Export-Package: com.fasterxml.jackson.dataformat.cbor;uses:="com.fasterxml.jackson.core,com.fasterxml.jackson.core.base,com.fasterxml.jackson.core.format,com.fasterxml.jackson.core.io,com.fasterxml.jackson.core.json,com.fasterxml.jackson.core.sym,com.fasterxml.jackson.core.util";version="2.12.3",com.fasterxml.jackson.dataformat.cbor.databind;uses:="com.fasterxml.jackson.core,com.fasterxml.jackson.databind,com.fasterxml.jackson.databind.cfg,com.fasterxml.jackson.dataformat.cbor";version="2.12.3"

Implementation-Build-Date: 2021-04-12 18:37:39+0000

Implementation-Title: Jackson dataformat: CBOR

Implementation-Vendor: FasterXML
Implementation-Vendor-Id: com.fasterxml.jackson.dataformat
Implementation-Version: 2.12.3
Import-Package: com.fasterxml.jackson.core;version="[2.12,3)",com.fasterxml.jackson.core.base;version="[2.12,3)",com.fasterxml.jackson.core.format;version="[2.12,3)",com.fasterxml.jackson.core.io;version="[2.12,3)",com.fasterxml.jackson.core.json;version="[2.12,3)",com.fasterxml.jackson.core.sym;version="[2.12,3)",com.fasterxml.jackson.core.util;version="[2.12,3)",com.fasterxml.jackson.databind;version="[2.12,3)",com.fasterxml.jackson.databind.cfg;version="[2.12,3)",com.fasterxml.jackson.dataformat.cbor;version="[2.12,3)"
Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.7))"
Specification-Title: Jackson dataformat: CBOR
Specification-Vendor: FasterXML
Specification-Version: 2.12.3
Tool: Bnd-5.1.1.202006162103
X-Compile-Source-JDK: 1.7
X-Compile-Target-JDK: 1.7

Found in path(s):

* /opt/cola/permits/1300428363_1649932293.82/0/jackson-dataformat-cbor-2-12-3-jar/META-INF/MANIFEST.MF

1.84 junit-platform-junit-platform-engine 1.8.2

1.84.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

- * **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
- * **b)** in the case of each subsequent Contributor:
 - * **i)** changes to the Program, and
 - * **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

****b)**** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

****i)**** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

****ii)**** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

****iii)**** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

****iv)**** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

****3.2)**** When the Program is Distributed as Source Code:

****a)**** it must be made available under this Agreement, or if the Program ****i)**** is combined with other material in a separate file or files made available under a Secondary License, and ****ii)**** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

****b)**** a copy of this Agreement must be included with each copy of the Program.

****3.3)**** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts

or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: ****a)**** promptly notify the Commercial Contributor in writing of such claim, and ****b)**** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or

conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.85 yaml-for-go v2.4.0

1.85.1 Available under license :

Copyright 2011-2016 Canonical Ltd.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The following files were ported to Go from C files of libyaml, and thus are still covered by their original copyright and license:

apic.go
emitterc.go
parserc.go
readerc.go
scannerc.go
writerc.go
yamlh.go
yamlprivateh.go

Copyright (c) 2006 Kirill Simonov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.86 reactive-streams v1.0.3

1.86.1 Available under license :

Copyright Statement for Contributions to the Reactive Streams Project

I hereby represent that all present, past and future contributions I make to the Reactive Streams project (which includes all repositories owned by the reactive-streams github organization) are governed by the Creative Commons Zero 1.0 Universal copyright statement, placing my contributions in the public domain. This entails that to the extent possible under law I waive all copyright and related or neighboring rights to the code or documents I contribute. I also represent that I have the authority to perform the above waiver with respect to the entirety of my contributions.

The text of the copyright statement is included in the COPYING file at the root of the reactive-streams repository at <https://github.com/reactive-streams/reactive-streams-jvm/blob/master/COPYING>.

Underwriting parties:

github name | Real Name, Email Address used for git commits, Company

rkuhn | Roland Kuhn, rk@rkuhn.info, Typesafe Inc.
benjchristensen | Ben Christensen, benjchristensen@gmail.com, Netflix Inc.
viktorklang | Viktor Klang, viktorklang@gmail.com, Typesafe Inc.
smaldini | Stephane Maldini, stephane.maldini@gmail.com, Pivotal Software Inc.
savulchik | Stanislav Savulchik, s.savulchik@gmail.com
ktoso | Konrad Malawski, konrad.malawski@project13.pl, Typesafe Inc.
ouertani | Slim Ouertani, ouertani@gmail.com
2m | Martynas Mickevicius, mmartynas@gmail.com, Typesafe Inc.
ldaley | Luke Daley, luke.daley@gradleware.com, Gradleware Inc.
colingodsey | Colin Godsey, crgodsey@gmail.com, MediaMath Inc.
davidmoten | Dave Moten, davidmoten@gmail.com
briantopping | Brian Topping, brian.topping@gmail.com, Mauswerks LLC
rstoyanchev | Rossen Stoyanchev, rstoyanchev@pivotal.io, Pivotal
BjornHamels | Bjrn Hamels, bjorn@hamels.nl
JakeWharton | Jake Wharton, jakewharton@gmail.com
anthonyvdotbe | Anthony Vanelverdinghe, anthonyv.be@outlook.com
seratch | Kazuhiro Sera, seratch@gmail.com, SmartNews, Inc.

akarnokd | David Karnok, akarnokd@gmail.com
egetman | Evgeniy Getman, getman.eugene@gmail.com
patriknw | Patrik Nordwall, patrik.nordwall@gmail.com, Lightbend Inc
angelsanz | ngel Sanz, angelsanz@users.noreply.github.com
shenghaiyang | , shenghaiyang@aliyun.com
kiiadi | Kyle Thomson, kylthoms@amazon.com, Amazon.com
jroper | James Roper, james@jazzy.id.au, Lightbend Inc.
olegdokuka | Oleh Dokuka, shadowgun@i.ua, Netifi Inc.
Scottmitch | Scott Mitchell, scott_mitchell@apple.com, Apple Inc.
retronym | Jason Zaugg, jzaugg@gmail.com, Lightbend Inc.
Creative Commons Legal Code

CC0 1.0 Universal

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS DOCUMENT DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE USE OF THIS DOCUMENT OR THE INFORMATION OR WORKS PROVIDED HEREUNDER, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM THE USE OF THIS DOCUMENT OR THE INFORMATION OR WORKS PROVIDED HEREUNDER.

Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

- i. the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work;
- ii. moral rights retained by the original author(s) and/or performer(s);
- iii. publicity and privacy rights pertaining to a person's image or likeness depicted in a Work;
- iv. rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below;
- v. rights protecting the extraction, dissemination, use and reuse of data in a Work;
- vi. database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive); and
- vii. other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non transferable, non sublicensable, non exclusive, irrevocable and unconditional license to exercise Affirmer's Copyright and

Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

- a. No trademark or patent rights held by Affirmer are waived, abandoned, surrendered, licensed or otherwise affected by this document.
- b. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law.
- c. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work.
- d. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.

Licensed under Public Domain (CC0)

To the extent possible under law, the person who associated CC0 with this code has waived all copyright and related or neighboring rights to this code.

You should have received a copy of the CC0 legalcode along with this work. If not, see <http://creativecommons.org/publicdomain/zero/1.0/>.

1.87 checker-qual 3.15.0

1.87.1 Available under license :

Checker Framework qualifiers

Copyright 2004-present by the Checker Framework developers

MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.88 allegro-bigcache v2.2.5

1.88.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.89 testcontainers-::-localstack 1.17.3

1.89.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the

Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.90 joda-time v2.10.2

1.90.1 Available under license :

=====

= NOTICE file corresponding to section 4d of the Apache License Version 2.0 =

=====

This product includes software developed by
Joda.org (<https://www.joda.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.91 aws-java-sdk-:::-metrics-interface 2.17.52

1.91.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by
Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.92 error_prone_annotations 2.3.4

1.92.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2014 The Error Prone Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/concurrent/UnlockMethod.java

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/concurrent/LockMethod.java

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/NoAllocation.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 The Error Prone Authors.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/RestrictedApi.java

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/DoNotMock.java

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/CompatibleWith.java

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/FormatMethod.java

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/FormatString.java

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/MustBeClosed.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017 The Error Prone Authors.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/CheckReturnValue.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/concurrent/GuardedBy.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/OverridingMethodsMustInvokeSuper.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/DoNotCall.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Error Prone Authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/SuppressPackageLocation.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/ForOverride.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/IncompatibleModifiers.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/concurrent/LazyInit.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-jar/com/google/errorprone/annotations/RequiredModifiers.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-


```
jar/com/google/errorprone/annotations/Var.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-
jar/com/google/errorprone/annotations/CanIgnoreReturnValue.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-
jar/com/google/errorprone/annotations/CompileTimeConstant.java
* /opt/cola/permits/1206718612_1632455182.54/0/error-prone-annotations-2-3-4-sources-9-
jar/com/google/errorprone/annotations/Immutable.java
```

1.93 dgryski/go-rendezvous 20200823-snapshot-9f7001d1

1.93.1 Available under license :

The MIT License (MIT)

Copyright (c) 2017-2020 Damian Gryski <damian@gryski.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.94 kotlin-scripting-jvm 1.3.50

1.94.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to

You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.95 junit-jupiter-junit-jupiter-engine 5.8.2

1.95.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and

* **b)** in the case of each subsequent Contributor:

* **i)** changes to the Program, and

* **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable),

including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1**** If a Contributor Distributes the Program in any form, then:

****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

****b)**** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

****i)**** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

****ii)**** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

****iii)**** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

****iv)**** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

****3.2**** When the Program is Distributed as Source Code:

****a)**** it must be made available under this Agreement, or if the Program ****i)**** is combined with other material in a separate file or files made available under a Secondary License, and ****ii)**** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

****b)**** a copy of this Agreement must be included with each copy of the Program.

****3.3**** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: ****a)**** promptly notify the Commercial Contributor in writing of such claim, and ****b)**** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the

Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.96 rxjava 2.1.14

1.96.1 Available under license :

No license file was found, but licenses were detected in source scan.

/**

* Copyright (c) 2016-present, RxJava Contributors.

* <p>

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in

* compliance with the License. You may obtain a copy of the License at


```
* <p>
* http://www.apache.org/licenses/LICENSE-2.0
* <p>
* Unless required by applicable law or agreed to in writing, software distributed under the License is
* distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
or implied. See
* the License for the specific language governing permissions and limitations under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableOnBackpressureError.java
No license file was found, but licenses were detected in source scan.
```

```
/**
* Copyright (c) 2016-present, RxJava Contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/schedulers/ScheduledDirectTask.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/schedulers/SchedulerWhen.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/schedulers/ScheduledDirectPeriodicTask.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/schedulers/AbstractDirectTask.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/schedulers/InstantPeriodicTask.java
No license file was found, but licenses were detected in source scan.
```

```
/**
* Copyright (c) 2016-present, RxJava Contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
```

- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/schedulers/SchedulerPoolFactory.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/BackpressureOverflowStrategy.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/annotations/package-info.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/schedulers/ComputationScheduler.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/disposables/package-info.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/subjects/package-info.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/package-info.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observables/package-info.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/processors/package-info.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/schedulers/NewThreadScheduler.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/exceptions/CompositeException.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableGroupJoin.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableJoin.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/schedulers/IOScheduler.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/functions/package-info.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowablePublishMulticast.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableGroupJoin.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/subscribers/package-info.java
- * /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/schedulers/package-info.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/schedulers/TrampolineScheduler.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observers/package-info.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/flowables/package-info.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/plugins/package-info.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/parallel/package-info.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/exceptions/package-info.java
No license file was found, but licenses were detected in source scan.

```
/**  
 * Copyright (c) 2016-present, RxJava Contributors.  
 *  
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in  
 * compliance with the License. You may obtain a copy of the License at  
 *  
 * http://www.apache.org/licenses/LICENSE-2.0  
 *  
 * Unless required by applicable law or agreed to in writing, software distributed under the License is  
 * distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express  
 * or implied. See  
 * the License for the specific language governing permissions and limitations under the License.  
 */
```

Found in path(s):

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/parallel/ParallelMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/schedulers/ImmediateThinScheduler.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observers/LambdaConsumerIntrospection.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/annotations/SchedulerSupport.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableDistinct.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleEquals.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableBlockingSubscribe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableIgnoreElements.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/util/BlockingIgnoringReceiver.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeFlatMapCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleDoOnError.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/exceptions/MissingBackpressureException.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeSwitchIfEmptySingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/observers/DeferredScalarObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableGroupBy.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableCount.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableReduceSeedSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableFlatMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/util/EmptyComponent.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSampleTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableMergeIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableSkipWhile.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeDoAfterSuccess.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeIgnoreElement.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableTakeUntilPredicate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleDoOnDispose.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableRetryWhen.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscriptions/ScalarSubscription.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableWindowBoundarySupplier.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/parallel/ParallelFromPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleTimer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleDelayWithObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableSampleTimed.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableCache.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/mixed/ObservableConcatMapCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/util/LinkedList.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservablePublish.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/functions/BiConsumer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeSwitchIfEmpty.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableDoFinally.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/BlockingObservableMostRecent.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSequenceEqualSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableFromPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/functions/Cancellable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableWithLatestFromMany.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableSequenceEqual.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableFromUnsafeSource.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableDoAfterNext.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableFlatMapMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableOnErrorNext.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableHide.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableRefCount.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableRetryPredicate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/disposables/SubscriptionDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/fuseable/SimpleQueue.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableTimeoutTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscriptions/AsyncSubscription.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/functions/Consumer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/ObservableTransformer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableDistinctUntilChanged.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableConcatMapPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeHide.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/observers/LambdaObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/mixed/ObservableSwitchMapCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/schedulers/NewThreadWorker.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/schedulers/NonBlockingThread.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeDelay.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableScanSeed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableCreate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleFlatMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSkipLast.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/mixed/FlowableConcatMapCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/MaybeObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableCache.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/SingleOnSubscribe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/mixed/ObservableConcatMapSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableEmpty.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableAll.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableWithLatestFrom.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableFromArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableCountSingle.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableReplay.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observers/ResourceMaybeObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/fuseable/FuseToObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/Single.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/BackpressureStrategy.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/Maybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/AbstractMaybeWithUpstream.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSingleSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observers/ResourceObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/disposables/CompositeDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableDefer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableFromCallable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/disposables/CancellableDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleJust.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableMergeWithCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSamplePublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableRetryBiPredicate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableNever.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/CompletableOnSubscribe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/observers/ForEachWhileObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/MaybeOperator.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/SingleTransformer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/Completable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableLift.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeFilter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/util/ListAddBiConsumer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableRepeatUntil.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/parallel/ParallelFlatMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowablePublish.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/flowables/ConnectableFlowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableMerge.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/mixed/ObservableSwitchMapMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableZipIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableCreate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeFromAction.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObserverResourceWrapper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableRepeat.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/SingleObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableScalarXMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/ObservableQueueDrain.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/functions/LongConsumer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeIsEmptySingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeDetach.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeCache.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/QueueDrain.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableJust.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableRepeatWhen.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/fuseable/SimplePlainQueue.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableFromArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/functions/Predicate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableWindow.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableDistinct.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableElementAtSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/parallel/ParallelCollect.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/DeferredScalarDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeUsing.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleAmb.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/schedulers/SingleScheduler.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/BasicIntQueueDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableMergeWithSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/NotificationLite.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeCreate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableDetach.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/ErrorMessage.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableFromFuture.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/observers/DisposableSingleObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableTakeLast.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableEmpty.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableLimit.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableSkipLastTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableGenerate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/subjects/CompletableSubject.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableFromObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/functions/Function3.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/exceptions/ProtocolViolationException.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableDoFinally.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/subjects/UnicastSubject.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/FutureSingleObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableFromRunnable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableDelaySubscriptionOther.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableFilter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableScan.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableMaterialize.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeTimeoutPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/CompletableTransformer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableUsing.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableOnErrorComplete.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableConcat.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableBufferBoundarySupplier.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/observables/ConnectableObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableError.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/ArrayListSupplier.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/observers/DisposableMaybeObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscribers/BasicFuseableSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/fuseable/HasUpstreamPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableWithLatestFromMany.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/mixed/ScalarXMapZHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/schedulers/DisposeOnCancel.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/EndConsumerHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableFromIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableError.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableZip.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableDoAfterNext.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableScan.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableSequenceEqualSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableMergeWithSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/parallel/ParallelJoin.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/fuseable/ScalarCallable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeDefer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscriptions/SubscriptionHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/MaybeTransformer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleDetach.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableSubscribeOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableSwitchIfEmpty.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeFlatMapIterableObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleFlatMapIterableObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/disposables/Disposables.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleTimeout.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableConcatWithCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableDematerialize.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/mixed/ObservableConcatMapMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/operators/flowable/FlowableRepeatWhen.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableMapNotification.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/SingleEmitter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/exceptions/Exceptions.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/functions/BooleanSupplier.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableUnsubscribeOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableOnErrorReturn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/parallel/ParallelPeek.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscribers/BlockingSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableOnBackpressureBufferStrategy.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableTimer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableBufferTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/parallel/ParallelFromArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableSingleSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableUnsubscribeOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleFlatMapCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableScalarXMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleCache.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableReduceSeedSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscriptions/SubscriptionArbiter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableCountSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableFilter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscribers/DeferredScalarSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/annotations/BackpressureSupport.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/parallel/ParallelConcatMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableUsing.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscriptions/ArrayCompositeSubscription.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableConcatWithSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/FutureObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleNever.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableSwitchMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableWindowBoundarySupplier.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleDelayWithSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/disposables/DisposableContainer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableDelay.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/ObservableOnSubscribe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableGroupBy.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableSkipUntil.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableZipIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/BlockingObservableIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/fuseable/FuseToMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/disposables/ListCompositeDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/CompletableObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableFlatMapCompletableCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/schedulers/SchedulerMultiWorkerSupport.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/MaybeSource.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableDoFinally.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/operators/completable/CompletableFromCallable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/subscribers/SafeSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableSwitchMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/queue/SpscLinkedListArrayQueue.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleZipArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/parallel/ParallelTransformer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/BlockingFlowableMostRecent.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/BlockingFlowableNext.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/BlockingMultiObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeTimeoutMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeCount.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleZipIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/SorterFunction.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/subjects/MaybeSubject.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableFlatMapPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/BasicFuseableObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/ConsumerSingleObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableTake.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableFlattenIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/disposables/RunnableDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableLift.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableSubscribeOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/observers/ResourceCompletableObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/parallel/ParallelFlowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/operators/maybe/MaybeEmpty.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleDelayWithCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableMergeWithMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableToListSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableIgnoreElements.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/BasicQueueDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/BackpressureHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableThrottleFirstTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableTakeUntilPredicate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableJoin.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/BlockingObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableConcatMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableDebounceTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/functions/ObjectHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/observers/DisposableCompletableObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableTakeWhile.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableConcatArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableFromPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/AppendOnlyLinkedList.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableConcatMapEagerPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscribers/InnerQueuedSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableRetryWhen.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableSkip.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeContains.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/operators/observable/ObservableBufferBoundarySupplier.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableSwitchIfEmpty.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableTimeout.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeFlatMapIterableFlowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/InnerQueuedObserverSupport.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableSubscribeOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeObserveOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/SuppressAnimalSniffer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeToObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/DisposableLambdaObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableToList.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableTimeout.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/flowables/GroupedFlowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/subscribers/SerializedSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/functions/Function8.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableLastSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleDoOnSubscribe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableInterval.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/Flowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeIsEmpty.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableFromFuture.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscribers/InnerQueuedSubscriberSupport.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/schedulers/RxThreadFactory.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/functions/BiPredicate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/BlockingObservableNext.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableRangeLong.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeFlatMapSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableInterval.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableLift.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSerialized.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/schedulers/TestScheduler.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableRangeLong.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSkipLastTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableBlockingSubscribe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeUnsafeCreate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleDoOnEvent.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableMergeDelayErrorArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableConcatWithMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleLift.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableConcatWithCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableZip.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscribers/StrictSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableConcatMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/ObservableSource.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableDelay.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/subscribers/DisposableSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeErrorCallable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableIntervalRange.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeFilterSingle.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableWindowTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableSkipUntil.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/parallel/ParallelFailureHandling.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/exceptions/OnErrorNotImplementedException.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/subjects/AsyncSubject.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybePeek.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableThrottleLatest.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableTakeWhile.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeOnErrorReturn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeIgnoreElementCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeFromCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableIntervalRange.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleDoAfterSuccess.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/disposables/Disposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscriptions/BasicIntQueueSubscription.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableToFlowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeFromFuture.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscriptions/BooleanSubscription.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSingleMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/processors/MulticastProcessor.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeToPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableAllSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableReduce.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/subjects/PublishSubject.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/Observer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableFromAction.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/functions/Function6.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/SingleOperator.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/BlockingFlowableIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableOnBackpressureBuffer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/mixed/ObservableSwitchMapSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/fuseable/QueueDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableLastMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observers/DisposableObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableCreate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableTimer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/queue/MpscLinkedQueue.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/fuseable/HasUpstreamMaybeSource.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableBufferTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableTakePublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/FlowableOperator.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/parallel/ParallelSortedJoin.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/subscribers/TestSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableMapNotification.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableCollectSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableLastSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleDoOnSuccess.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/operators/flowable/FlowableToList.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableFlatMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableFlattenIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableGenerate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableReduceWithSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleToObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableTakeUntil.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeFromRunnable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/processors/SerializedProcessor.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/annotations/NonNull.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableDoOnEach.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/SubscriberCompletableObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeUnsubscribeOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/subjects/Subject.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableDetach.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/FlowableTransformer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableOnBackpressureDrop.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/disposables/ReferenceDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableInternalHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/processors/BehaviorProcessor.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableFlatMapSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableLastMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableObserveOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableFlatMapSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/operators/observable/ObservableFlatMapMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/CompletableConverter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/mixed/FlowableConcatMapMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableSerialized.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/HalfSerializer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeOnErrorNext.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/parallel/ParallelFilterTry.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableBuffer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableConcatArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableThrottleLatest.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/functions/BiFunction.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/BlockingBaseObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/BlockingFirstObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableRetryBiPredicate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableSequenceEqual.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableRefCount.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableAny.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/fuseable/QueueSubscription.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/annotations/BackpressureKind.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableDematerialize.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/Emitter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/disposables/FutureDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeMergeArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleToFlowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/functions/Function4.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeFlatMapBiSelector.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/disposables/SequentialDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableMergeWithMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/subscribers/ResourceSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSkip.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeNever.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/observers/BiConsumerSingleObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableBuffer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableMergeArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableBufferBoundary.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableConcatWithSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleUnsubscribeOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/util/ConnectConsumer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/MaybeEmitter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeEqualSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableAutoConnect.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/parallel/ParallelFilter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeTakeUntilPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableTimeInterval.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/disposables/ActionDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableHide.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/exceptions/UndeliverableException.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableSampleWithObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeCallbackObserver.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableTakeLastOne.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/AbstractFlowableWithUpstream.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableElementAtSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableReduceMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/SingleConverter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/util/HashMapSupplier.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableDebounce.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableUsing.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleFromUnsafeSource.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/disposables/DisposableHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/FlowableEmitter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/observers/EmptyCompletableObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableCombineLatest.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/processors/ReplayProcessor.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeAmb.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableNever.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/observers/QueueDrainObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/processors/AsyncProcessor.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableTimeoutTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableRange.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableFlatMapCompletableCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableFromSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableOnBackpressureLatest.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableDistinctUntilChanged.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableNever.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeDoOnEvent.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeFromSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/mixed/FlowableConcatMapSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscribers/BlockingBaseSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/subjects/SerializedSubject.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleDelay.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableDoOnLifecycle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscribers/SinglePostCompleteSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableCollect.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableMaterialize.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/plugins/RxJavaPlugins.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableMapPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableElementAt.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/functions/Function5.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/FlowableOnSubscribe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableDoOnLifecycle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeDelayWithCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/util/AtomicThrowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSkipWhile.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableFromPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/functions/Function7.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscriptions/DeferredScalarSubscription.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableMap.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleObserveOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/observers/InnerQueuedObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableTakeUntil.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableWindowBoundary.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/util/MergerBiFunction.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableHide.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeError.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableWindowTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleResumeNext.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscribers/ForEachWhileSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observers/ResourceSingleObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableFlatMapCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableDoOnEach.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleHide.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/schedulers/Schedulers.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableAmb.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/mixed/FlowableSwitchMapCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableAnySingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableElementAtMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/parallel/ParallelReduceFull.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/CompletableOperator.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableDelaySubscriptionOther.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableCount.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeDelayOtherPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableMergeWithCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableCombineLatest.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableReduceWithSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableRetryPredicate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableError.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleDoAfterTerminate.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableBufferExactBoundary.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/functions/Action.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/schedulers/ScheduledRunnable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableWindow.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableWindowBoundarySelector.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservablePublishSelector.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleFromCallable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/disposables/ArrayCompositeDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableSkipLast.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableConcatIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableThrottleFirstTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/util/QueueDrainHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeJust.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableFromUnsafeSource.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableToObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableEmpty.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/subjects/ReplaySubject.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observers/DefaultObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableDefer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableToListSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/fuseable/package-info.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableRepeat.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableAutoConnect.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableFlatMapCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableObserveOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableCache.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/fuseable/ConditionalSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscriptions/BasicQueueSubscription.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableAnySingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableInternalHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableToSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableWindowBoundarySelector.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleUsing.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observers/SerializedObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/Observable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observables/GroupedObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/fuseable/FuseToFlowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/functions/IntFunction.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleFlatMapMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableTake.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/operators/flowable/FlowableFromIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/parallel/ParallelDoOnNextTry.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/schedulers/Timed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/CallbackCompletableObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/OpenHashSet.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableOnErrorReturn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableRepeatUntil.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeToSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeFlatMapSingleElement.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeSubscribeOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableElementAt.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/fuseable/HasUpstreamCompletableSource.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableConcatWithMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/ObservableOperator.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/annotations/CheckReturnValue.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableResumeNext.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableErrorSupplier.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableAmb.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleDefer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableConcatMapEager.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/subjects/SingleSubject.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/subscribers/DefaultSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeTimer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableWithLatestFrom.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/ObservableEmitter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleOnErrorReturn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableFromCallable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/functions/Function9.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/fuseable/QueueFuseable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableDisposeOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscribers/SubscriberResourceWrapper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/Scheduler.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscribers/BlockingFirstSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/Notification.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/fuseable/HasUpstreamObservableSource.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeDelaySubscriptionOtherPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/parallel/ParallelFlowableConverter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeDoFinally.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleSubscribeOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/subjects/BehaviorSubject.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeToFlowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleTakeUntil.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/fuseable/HasUpstreamSingleSource.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableJust.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableOnErrorNext.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleContains.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/parallel/ParallelMapTry.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/BlockingObservableLatest.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableCollect.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/operators/maybe/MaybeZipIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/mixed/FlowableSwitchMapMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableScanSeed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/observers/BaseTestConsumer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleDoFinally.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableAllSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableFromObservable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/FlowableSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeFlatten.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableTimeout.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/schedulers/ExecutorScheduler.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/BlockingHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableDebounceTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/MaybeConverter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/annotations/Nullable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscribers/QueueDrainSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeLift.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/CompletableSource.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableSingleMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/ResumeSingleObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/FlowableConverter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableTakeLastTimed.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscriptions/EmptySubscription.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/observers/SafeObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-

jar/io/reactivex/internal/util/VolatileSizeArrayList.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableConcatMapEager.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableBufferBoundary.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/observers/BlockingLastObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/disposables/SerialDisposable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableAll.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeConcatArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/annotations/Beta.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeConcatIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/subscribers/FutureSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/parallel/ParallelReduce.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/maybe/MaybeOnErrorComplete.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleError.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableDefer.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/BlockingFlowableLatest.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletableMergeDelayErrorIterable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/MaybeOnSubscribe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableTakeLastOne.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/observable/ObservableAny.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/completable/CompletablePeek.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableDebounce.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/single/SingleFlatMapIterableFlowable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/CompletableEmitter.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/operators/flowable/FlowableIgnoreElementsCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/SingleSource.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscribers/LambdaSubscriber.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/mixed/FlowableSwitchMapSingle.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableReduceMaybe.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/disposables/EmptyDisposable.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/parallel/ParallelRunOn.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/functions/Function.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableReplay.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableWindowBoundary.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/ObservableConverter.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableAmb.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleInternalHelper.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/processors/UnicastProcessor.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscribers/BasicFuseableConditionalSubscriber.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableRange.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleCreate.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableDelay.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableTimeInterval.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/processors/PublishProcessor.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/completable/CompletableDoOnEvent.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableBufferExactBoundary.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableTimer.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableTakeLast.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableTakeLastTimed.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/queue/SpSCArrayQueue.java

* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableObserveOn.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableDetach.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableIgnoreElementsCompletable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleDelayWithPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/annotations/Experimental.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/single/SingleFromPublisher.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/processors/FlowableProcessor.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/AbstractObservableWithUpstream.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeFlatMapNotification.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/subscribers/BlockingLastSubscriber.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/flowable/FlowableCollectSingle.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeZipArray.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeTakeUntilMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeFromCallable.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/observers/TestObserver.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/util/ExceptionHelper.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/functions/Functions.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/observable/ObservableElementAtMaybe.java
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-jar/io/reactivex/internal/operators/maybe/MaybeConcatArrayDelayError.java
No license file was found, but licenses were detected in source scan.

/**

* Copyright (c) 2016-present, RxJava Contributors.

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in
* compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

```
* Unless required by applicable law or agreed to in writing, software distributed under the License is
* distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
or implied. See
* the License for the specific language governing permissions and limitations under the License.
*/
/*
* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE
* Original location: https://github.com/JCTools/JCTools/blob/master/jctools-
core/src/main/java/org/jctools/util/Pow2.java
*/
```

Found in path(s):

```
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/internal/util/Pow2.java
```

No license file was found, but licenses were detected in source scan.

```
/**
```

```
* Copyright (c) 2016-present, RxJava Contributors.
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in
* compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software distributed under the License is
* distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
or implied. See
* the License for the specific language governing permissions and limitations under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1411866817_1662683809.7862647/0/rxjava-2-1-14-sources-1-
jar/io/reactivex/schedulers/SchedulerRunnableIntrospection.java
```

1.97 matttproud-golang-protobuf-extensions

v1.0.1

1.97.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright 2012 Matt T. Proud (matt.proud@gmail.com)

1.98 asm 9.1

1.98.1 Available under license :

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. Neither the name of the copyright holders nor the names of its

* contributors may be used to endorse or promote products derived from

* this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"

* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE

* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR

* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF

* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN

* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)

* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF

* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):

* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-

jar/org/objectweb/asm/signature/package.html

* /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/package.html

No license file was found, but licenses were detected in source scan.

// All rights reserved.

// Redistribution and use in source and binary forms, with or without

// modification, are permitted provided that the following conditions

// are met:

// 1. Redistributions of source code must retain the above copyright

// notice, this list of conditions and the following disclaimer.

// 2. Redistributions in binary form must reproduce the above copyright

// notice, this list of conditions and the following disclaimer in the

// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):

- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/TypePath.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Frame.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/RecordComponentWriter.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/AnnotationVisitor.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ModuleVisitor.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ModuleWriter.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/CurrentFrame.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/AnnotationWriter.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ConstantDynamic.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ClassWriter.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/signature/SignatureVisitor.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Attribute.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Opcodes.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Handle.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/MethodTooLargeException.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/signature/SignatureReader.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/TypeReference.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Context.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Label.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Type.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/signature/SignatureWriter.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/FieldVisitor.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Handler.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/SymbolTable.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ClassReader.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Edge.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/RecordComponentVisitor.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Constants.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/Symbol.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/FieldWriter.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/MethodWriter.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/MethodVisitor.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ClassTooLargeException.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ByteVector.java
- * /opt/cola/permits/1150153141_1627681655.27/0/asm-9-1-sources-1-jar/org/objectweb/asm/ClassVisitor.java

1.99 testcontainers-junit-jupiter-extension

1.17.3

1.99.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.100 ginkgo 1.16.4

1.100.1 Available under license :

The MIT License (MIT)

Copyright (c) 2016 Yasuhiro Matsumoto

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) Yasuhiro MATSUMOTO <mattn.jp@gmail.com>

MIT License (Expat)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013-2014 Onsi Fakhouri

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.101 docker-java-transport 3.2.13

1.101.1 Available under license :

Apache-2.0

1.102 jackson-annotations 2.12.5

1.102.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.103 junit-jupiter-junit-jupiter-api 5.8.2

1.103.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

- * **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
- * **b)** in the case of each subsequent Contributor:
 - * **i)** changes to the Program, and
 - * **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1**** If a Contributor Distributes the Program in any form, then:

* ****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* ****b)**** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

* ****i)**** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* ****ii)**** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

* ****iii)**** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

* ****iv)**** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

****3.2**** When the Program is Distributed as Source Code:

* ****a)**** it must be made available under this Agreement, or if the Program ****i)**** is combined with other material in a separate file or files made available under a Secondary License, and ****ii)**** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* ****b)**** a copy of this Agreement must be included with each copy of the Program.

****3.3**** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: ****a)**** promptly notify the Commercial Contributor in writing of such claim, and ****b)**** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That

Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the

Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.104 junit 4.12

1.104.1 Available under license :

JUnit

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

- a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
- b) in the case of each subsequent Contributor:

- i) changes to the Program, and

- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

- a) it complies with the terms and conditions of this Agreement; and
- b) its license agreement:
 - i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 - ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 - iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
 - iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

- a) it must be made available under this Agreement; and
- b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise

of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of

the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

1.105 project-lombok 1.18.20

1.105.1 Available under license :

Copyright (C) 2009-2021 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====
Licenses for included components:

org.ow2.asm:asm

org.ow2.asm:asm-analysis

org.ow2.asm:asm-commons

org.ow2.asm:asm-tree

org.ow2.asm:asm-util

ASM: a very small and fast Java bytecode manipulation framework

Copyright (c) 2000-2011 INRIA, France Telecom

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

rzwitserloot/com.zwitsersloot.cmdreader

Copyright 2010 Reinier Zwitsersloot.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

rzwitserloot/lombok.patcher

Copyright (C) 2009-2021 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.106 jna 5.8.0

1.106.1 Available under license :

/*

- * The contents of this file are subject to the Mozilla Public License Version 1.1
- * (the "License"); you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at <http://www.mozilla.org/MPL/>
- *
- * Software distributed under the License is distributed on an "AS IS" basis,
- * WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
- * for the specific language governing rights and limitations under the License.
- *
- * The Original Code is 'iText, a free JAVA-PDF library'.
- *
- * The Initial Developer of the Original Code is Bruno Lowagie. Portions created by
- * the Initial Developer are Copyright (C) 1999, 2000, 2001, 2002 by Bruno Lowagie.
- * All Rights Reserved.
- * Co-Developer of the code is Paulo Soares. Portions created by the Co-Developer
- * are Copyright (C) 2000, 2001, 2002 by Paulo Soares. All Rights Reserved.
- *
- * Contributor(s): all the names of the contributors are added in the source code

* where applicable.

*

* Alternatively, the contents of this file may be used under the terms of the

* LGPL license (the "GNU LIBRARY GENERAL PUBLIC LICENSE"), in which case the

* provisions of LGPL are applicable instead of those above. If you wish to

* allow use of your version of this file only under the terms of the LGPL

* License and not to allow others to use your version of this file under

* the MPL, indicate your decision by deleting the provisions above and

* replace them with the notice and other provisions required by the LGPL.

* If you do not delete the provisions above, a recipient may use your version

* of this file under either the MPL or the GNU LIBRARY GENERAL PUBLIC LICENSE.

*

* This library is free software; you can redistribute it and/or modify it

* under the terms of the MPL as stated above or under the terms of the GNU

* Library General Public License as published by the Free Software Foundation;

* either version 2 of the License, or any later version.

*

* This library is distributed in the hope that it will be useful, but WITHOUT

* ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS

* FOR A PARTICULAR PURPOSE. See the GNU Library general Public License for more

* details.

*

* If you didn't download this code from the following link, you should check if

* you aren't using an obsolete version:

* <http://www.lowagie.com/iText/>

*

* This class is generated based on a grammar file provided by SUN, and updated

* by Carsten Hammer. SUN's license agreement can be found at this URL:

* <http://java.sun.com/products/java-media/2D/samples/samples-license.html>

* See also the file sun.txt in directory com.lowagie.text.pdf

*/

Maven Ant Tasks

Copyright 2002-2011 The Apache Software Foundation

This product includes software developed at
 The Apache Software Foundation (<http://www.apache.org/>).

=====

=====

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from 'org.jfree' to 'clover.org.jfree'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

=====

=====

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the

library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from

such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in

themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the

Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under

Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on

the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to

refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing

and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED

TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Ant

Copyright 1999-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

The <sync> task is based on code Copyright (c) 2002, Landmark Graphics Corp that has been kindly donated to the Apache Software Foundation.

/*

* Apache License
* Version 2.0, January 2004
* <http://www.apache.org/licenses/>
*

* TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
*

* 1. Definitions.
*

* "License" shall mean the terms and conditions for use, reproduction,
* and distribution as defined by Sections 1 through 9 of this document.
*

* "Licensor" shall mean the copyright owner or entity authorized by
* the copyright owner that is granting the License.
*

* "Legal Entity" shall mean the union of the acting entity and all
* other entities that control, are controlled by, or are under common
* control with that entity. For the purposes of this definition,
* "control" means (i) the power, direct or indirect, to cause the
* direction or management of such entity, whether by contract or
* otherwise, or (ii) ownership of fifty percent (50%) or more of the
* outstanding shares, or (iii) beneficial ownership of such entity.
*

* "You" (or "Your") shall mean an individual or Legal Entity
* exercising permissions granted by this License.
*

* "Source" form shall mean the preferred form for making modifications,
* including but not limited to software source code, documentation
* source, and configuration files.
*

* "Object" form shall mean any form resulting from mechanical
* transformation or translation of a Source form, including but
* not limited to compiled object code, generated documentation,
*

* and conversions to other media types.

*

* "Work" shall mean the work of authorship, whether in Source or
* Object form, made available under the License, as indicated by a
* copyright notice that is included in or attached to the work
* (an example is provided in the Appendix below).

*

* "Derivative Works" shall mean any work, whether in Source or Object
* form, that is based on (or derived from) the Work and for which the
* editorial revisions, annotations, elaborations, or other modifications
* represent, as a whole, an original work of authorship. For the purposes
* of this License, Derivative Works shall not include works that remain
* separable from, or merely link (or bind by name) to the interfaces of,
* the Work and Derivative Works thereof.

*

* "Contribution" shall mean any work of authorship, including
* the original version of the Work and any modifications or additions
* to that Work or Derivative Works thereof, that is intentionally
* submitted to Licensor for inclusion in the Work by the copyright owner
* or by an individual or Legal Entity authorized to submit on behalf of
* the copyright owner. For the purposes of this definition, "submitted"
* means any form of electronic, verbal, or written communication sent
* to the Licensor or its representatives, including but not limited to
* communication on electronic mailing lists, source code control systems,
* and issue tracking systems that are managed by, or on behalf of, the
* Licensor for the purpose of discussing and improving the Work, but
* excluding communication that is conspicuously marked or otherwise
* designated in writing by the copyright owner as "Not a Contribution."

*

* "Contributor" shall mean Licensor and any individual or Legal Entity
* on behalf of whom a Contribution has been received by Licensor and
* subsequently incorporated within the Work.

*

* 2. Grant of Copyright License. Subject to the terms and conditions of
* this License, each Contributor hereby grants to You a perpetual,
* worldwide, non-exclusive, no-charge, royalty-free, irrevocable
* copyright license to reproduce, prepare Derivative Works of,
* publicly display, publicly perform, sublicense, and distribute the
* Work and such Derivative Works in Source or Object form.

*

* 3. Grant of Patent License. Subject to the terms and conditions of
* this License, each Contributor hereby grants to You a perpetual,
* worldwide, non-exclusive, no-charge, royalty-free, irrevocable
* (except as stated in this section) patent license to make, have made,
* use, offer to sell, sell, import, and otherwise transfer the Work,
* where such license applies only to those patent claims licensable
* by such Contributor that are necessarily infringed by their
* Contribution(s) alone or by combination of their Contribution(s)

* with the Work to which such Contribution(s) was submitted. If You
* institute patent litigation against any entity (including a
* cross-claim or counterclaim in a lawsuit) alleging that the Work
* or a Contribution incorporated within the Work constitutes direct
* or contributory patent infringement, then any patent licenses
* granted to You under this License for that Work shall terminate
* as of the date such litigation is filed.

*

* 4. Redistribution. You may reproduce and distribute copies of the
* Work or Derivative Works thereof in any medium, with or without
* modifications, and in Source or Object form, provided that You
* meet the following conditions:

*

* (a) You must give any other recipients of the Work or
* Derivative Works a copy of this License; and

*

* (b) You must cause any modified files to carry prominent notices
* stating that You changed the files; and

*

* (c) You must retain, in the Source form of any Derivative Works
* that You distribute, all copyright, patent, trademark, and
* attribution notices from the Source form of the Work,
* excluding those notices that do not pertain to any part of
* the Derivative Works; and

*

* (d) If the Work includes a "NOTICE" text file as part of its
* distribution, then any Derivative Works that You distribute must
* include a readable copy of the attribution notices contained
* within such NOTICE file, excluding those notices that do not
* pertain to any part of the Derivative Works, in at least one
* of the following places: within a NOTICE text file distributed
* as part of the Derivative Works; within the Source form or
* documentation, if provided along with the Derivative Works; or,
* within a display generated by the Derivative Works, if and
* wherever such third-party notices normally appear. The contents
* of the NOTICE file are for informational purposes only and
* do not modify the License. You may add Your own attribution
* notices within Derivative Works that You distribute, alongside
* or as an addendum to the NOTICE text from the Work, provided
* that such additional attribution notices cannot be construed
* as modifying the License.

*

* You may add Your own copyright statement to Your modifications and
* may provide additional or different license terms and conditions
* for use, reproduction, or distribution of Your modifications, or
* for any such Derivative Works as a whole, provided Your use,
* reproduction, and distribution of the Work otherwise complies with
* the conditions stated in this License.

- *
* 5. Submission of Contributions. Unless You explicitly state otherwise,
* any Contribution intentionally submitted for inclusion in the Work
* by You to the Licensor shall be under the terms and conditions of
* this License, without any additional terms or conditions.
* Notwithstanding the above, nothing herein shall supersede or modify
* the terms of any separate license agreement you may have executed
* with Licensor regarding such Contributions.
*
- * 6. Trademarks. This License does not grant permission to use the trade
* names, trademarks, service marks, or product names of the Licensor,
* except as required for reasonable and customary use in describing the
* origin of the Work and reproducing the content of the NOTICE file.
*
- * 7. Disclaimer of Warranty. Unless required by applicable law or
* agreed to in writing, Licensor provides the Work (and each
* Contributor provides its Contributions) on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
* implied, including, without limitation, any warranties or conditions
* of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
* PARTICULAR PURPOSE. You are solely responsible for determining the
* appropriateness of using or redistributing the Work and assume any
* risks associated with Your exercise of permissions under this License.
*
- * 8. Limitation of Liability. In no event and under no legal theory,
* whether in tort (including negligence), contract, or otherwise,
* unless required by applicable law (such as deliberate and grossly
* negligent acts) or agreed to in writing, shall any Contributor be
* liable to You for damages, including any direct, indirect, special,
* incidental, or consequential damages of any character arising as a
* result of this License or out of the use or inability to use the
* Work (including but not limited to damages for loss of goodwill,
* work stoppage, computer failure or malfunction, or any and all
* other commercial damages or losses), even if such Contributor
* has been advised of the possibility of such damages.
*
- * 9. Accepting Warranty or Additional Liability. While redistributing
* the Work or Derivative Works thereof, You may choose to offer,
* and charge a fee for, acceptance of support, warranty, indemnity,
* or other liability obligations and/or rights consistent with this
* License. However, in accepting such obligations, You may act only
* on Your own behalf and on Your sole responsibility, not on behalf
* of any other Contributor, and only if You agree to indemnify,
* defend, and hold each Contributor harmless for any liability
* incurred by, or claims asserted against, such Contributor by reason
* of your accepting any such warranty or additional liability.
*
- * END OF TERMS AND CONDITIONS

*
 * APPENDIX: How to apply the Apache License to your work.
 *
 * To apply the Apache License to your work, attach the following
 * boilerplate notice, with the fields enclosed by brackets "[]"
 * replaced with your own identifying information. (Don't include
 * the brackets!) The text should be enclosed in the appropriate
 * comment syntax for the file format. We also recommend that a
 * file or class name and description of purpose be included on the
 * same "printed page" as the copyright notice for easier
 * identification within third-party archives.
 *
 * Copyright [yyyy] [name of copyright owner]
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * <http://www.apache.org/licenses/LICENSE-2.0>
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

W3C SOFTWARE NOTICE AND LICENSE

<http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date

changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission.

Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

This license came from: <http://www.megginson.com/SAX/copying.html>
However please note future versions of SAX may be covered
under <http://saxproject.org/?selected=pd>

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05

=====
=====
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from 'com.keypoint/org.jfree' to 'clover.com.keypoint/clover.org.jfree'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

=====
=====

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for

you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many

libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is

included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany

it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by

the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively

convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the

Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Lang

Copyright 2001-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

/*

* The Apache Software License, Version 1.1

*

* Copyright (c) 2000-2003 The Apache Software Foundation. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

*

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

*

* 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgement:

* "This product includes software developed by the

* Apache Software Foundation (<http://www.apache.org/>)."

* Alternately, this acknowledgement may appear in the software itself, if and wherever such third-party acknowledgements normally appear.

*

* 4. The names "Ant" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

*

* 5. Products derived from this software may not be called "Apache" nor may "Apache" appear in their names without prior written permission of the Apache Group.

*

* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* =====

*

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* <<http://www.apache.org/>>.
*/

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

- a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or
- b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

- a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

- a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the Combined Work with a copy of the GNU GPL and this license

document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

MOZILLA PUBLIC LICENSE

Version 1.1

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally

accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.

The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification

has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims.

If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.

If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely

to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.

You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.

You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to

statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.

Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.

Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.

If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF

DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as

by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United

States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is _____.

The Initial Developer of the Original Code is _____.
Portions created by _____ are Copyright (C) _____
_____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[_____] License"), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [_____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [_____] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [_____] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

/*

Copyright (c) 2000, Derek Petillo
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Praxis Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Public Domain Dedication

This license is acceptable for Free Cultural Works.

Copyright-Only Dedication (based on United States law) or Public Domain Certification

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

Open Source License for the overlibmws Package

1. License coverage

Note that this license only covers the script library (javascript core and plugin modules) and not any supporting material such as the overlibmws website or its online documentation and support files. You may not reproduce the website or its online material without explicit written permission from the author, but can freely incorporate scripts and procedures which are demonstrated in that material into your own HTML or XML documents.

2. License (Artistic)

Preamble

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:

"Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)

"Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.

You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:

- place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as ftp.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

- use the modified Package only within your corporation or organization.

- rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

- make other distribution arrangements with the Copyright Holder.

You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

- distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

- accompany the distribution with the machine-readable source of the Package with your modifications.

- accompany any non-standard executables with their corresponding Standard Version executables, giving the non-standard executables non-standard names, and clearly documenting the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

- make other distribution arrangements with the Copyright Holder.

You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own.

The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whomever generated them, and may be sold commercially, and may be aggregated with this Package.

C or perl subroutines supplied by you and linked into this Package shall not be considered part of this Package.

The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND

FITNESS FOR A PARTICULAR PURPOSE.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

=====
=====

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from 'it.unimi.dsi.fastutil' to 'clover.it.unimi.dsi.fastutil'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

=====
=====

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in

non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does

and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based

on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and

therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License.

Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the

user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any

particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU

Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Javolution - Java(TM) Solution for Real-Time and Embedded Systems
Copyright (c) 2006, Javolution (<http://javolution.org>)
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The libffi source distribution contains certain code that is not part of libffi, and is only used as tooling to assist with the building and testing of libffi. This includes the msvcc.sh script used to wrap the Microsoft compiler with GNU compatible command-line options, make_sunver.pl, and the libffi test code distributed in the testsuite/libffi.bhaible directory. This code is distributed with

libffi for the purpose of convenience only, and libffi is in no way derived from this code.

msvcc.sh an testsuite/libffi.bhaible are both distributed under the terms of the GNU GPL version 2, as below.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy,

distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the

Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding

those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author  
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.  
This is free software, and you are welcome to redistribute it  
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

libffi - Copyright (c) 1996-2020 Anthony Green, Red Hat, Inc and others.
See source files for details.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache Commons BeanUtils
Copyright 2000-2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

=====
=====

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed
their package names from 'org.apache.velocity' to 'clover.org.apache.velocity'. This was necessary to
avoid potential name conflicts during instrumentation of a code using the original library when using Clover.
No source code of the original library was modified.

=====
=====

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including

the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each

Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

\$Id: LICENSE.txt,v 1.1.1.1 2004/07/01 13:59:13 jvanzyl Exp \$

Copyright 2002 (C) The Codehaus. All Rights Reserved.

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name "classworlds" must not be used to endorse or promote products derived from this Software without prior written permission of The Codehaus. For written permission, please contact bob@codehaus.org.
4. Products derived from this Software may not be called "classworlds" nor may "classworlds" appear in their names without prior written permission of The Codehaus. "classworlds" is a registered trademark of The Codehaus.
5. Due credit should be given to The Codehaus.
(<http://classworlds.codehaus.org/>).

THIS SOFTWARE IS PROVIDED BY THE CODEHAUS AND CONTRIBUTORS ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE CODEHAUS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

=====
=====
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from the 'antlr.*' to the 'clover.antlr.*'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover.
No source code of the original library was modified.

=====
=====
SOFTWARE RIGHTS

ANTLR 1989-2006 Developed by Terence Parr
Partially supported by University of San Francisco & jGuru.com

We reserve no legal rights to the ANTLR--it is fully in the public domain. An individual or company may do whatever they wish with source code distributed with ANTLR or the code generated by ANTLR, including the incorporation of ANTLR, or its output, into commercial software.

We encourage users to develop software with ANTLR. However, we do ask that credit is given to us for developing ANTLR. By "credit", we mean that if you use ANTLR or incorporate any source code into one of your programs (commercial product, research project, or otherwise) that you acknowledge this fact somewhere in the documentation, research report, etc... If you like ANTLR and have developed a nice tool with the output, please mention that you developed it using ANTLR. In addition, we ask that the headers remain intact in our source code. As long as these guidelines are kept, we expect to continue enhancing this system and expect to make other tools available as they are completed.

The primary ANTLR guy:

Terence Parr
parrt@cs.usfca.edu

parrr@antlr.org

=====
=====
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package name from 'com.google.common' to 'clover.com.google.common'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

=====
=====

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

=====
=====

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed
their package name from 'com.google.json' to 'clover.com.google.json'. This was necessary to
avoid potential name conflicts during instrumentation of a code using the original library when using Clover.
No source code of the original library was modified.

=====
=====

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or

documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,

work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====
=====

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package name from 'com.lowagie' to 'clover.com.lowagie'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover.

No source code of the original library was modified.

=====
=====

MOZILLA PUBLIC LICENSE

Version 1.1

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a

Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by

that Contributor.

3. Distribution Obligations.

3.1. Application of License.

The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims.

If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will

know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.

If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.

You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included

in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.

You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.

Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.

Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.

If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF

LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is _____.

The Initial Developer of the Original Code is _____.
Portions created by _____ are Copyright (C) _____
_____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[] License"), in which case the provisions of [] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

Apache Velocity

Copyright (C) 2000-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

onoMnApeShTpQtDJbcUgJTIFONPQeUndIgfQWWNNddIwBl
mi2Kp5RjfhIJdGCSO<bOTNof2KNxm9KCi5lxEyKI9BJW3p
qOPQUXpopOopMMPqnPnXXQPNOPNRnqQNQqStwVxuQSTtVW
UrwSUSSTVwxWSXNmrrpnmqmUUnpsvpntsmmmmmUUnpsvp
ntsmmmmmUUFmbkWJlroZbW4bsbilmjbkqUUnmmmm
Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name "DOM4J" must not be used to endorse or promote products derived from this Software without prior written permission of MetaStuff, Ltd. For written permission, please contact dom4j-info@metastuff.com.
4. Products derived from this Software may not be called "DOM4J" nor may "DOM4J" appear in their names without prior written permission of MetaStuff, Ltd. DOM4J is a registered trademark of MetaStuff, Ltd.
5. Due credit should be given to the DOM4J Project - <http://www.dom4j.org>

THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*--

\$Id: LICENSE.txt,v 1.11 2004/02/06 09:32:57 jhunter Exp \$

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.
3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.
4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the
JDOM Project (<http://www.jdom.org/>)."

Alternatively, the acknowledgment may be graphical using the logos available at <http://www.jdom.org/images/logos>.

THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and

Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <<http://www.jdom.org/>>.

*/

From: <http://www.json.org/license.html>

=====
=====
Copyright (c) 2002 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

SPDX-License-Identifier: Apache-2.0 OR LGPL-2.1

Java Native Access (JNA) is licensed under the LGPL, version 2.1 or later, or (from version 4.0 onward) the Apache License, version 2.0.

You can freely decide which license you want to apply to the project.

You may obtain a copy of the LGPL License at:

<http://www.gnu.org/licenses/licenses.html>

A copy is also included in the downloadable source code package containing JNA, in file "LGPL2.1", under the same directory as this file.

You may obtain a copy of the Apache License at:

<http://www.apache.org/licenses/>

A copy is also included in the downloadable source code package containing JNA, in file "AL2.0", under the same directory as this file.

Commercial support may be available, please e-mail twall[at]users[dot]sf[dot]net.

=====
=====
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package name from 'org.apache.commons' to 'clover.org.apache.commons'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

=====
=====
Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but

not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their

Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with

the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright (c) 2005 - 2009 Taras Puchko
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/******

* CruiseControl, a Continuous Integration Toolkit
* Copyright (c) 2001-2003, ThoughtWorks, Inc.
* 651 W Washington Ave. Suite 500
* Chicago, IL 60661 USA
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* + Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* + Redistributions in binary form must reproduce the above
* copyright notice, this list of conditions and the following
* disclaimer in the documentation and/or other materials provided
* with the distribution.
*
* + Neither the name of ThoughtWorks, Inc., CruiseControl, nor the
* names of its contributors may be used to endorse or promote
* products derived from this software without specific prior
* written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
* A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR
* CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
* PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
* LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
* NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*****/

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent

to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,

excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any

risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

http://www.atlassian.com/dms/wac/Atlassian_EULA_4-2.pdf

GNU LIBRARY GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some
specially designated Free Software Foundation software, and to any
other libraries whose authors decide to use it. You can use it for
your libraries, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if
you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link a program with the library, you must provide
complete object files to the recipients so that they can relink them
with the library, after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright

the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact

all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest

your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License.

Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies,

or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU

FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

/*--

\$Id: LICENSE.txt,v 1.11 2004/02/06 09:32:57 jhunter Exp \$

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.
3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.
4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the
JDOM Project (<http://www.jdom.org/>)."

Alternatively, the acknowledgment may be graphical using the logos available at <http://www.jdom.org/images/logos>.

THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT

CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <<http://www.jdom.org/>>.

*/

Indiana University Extreme! Lab Software License

Version 1.1.1

Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the Indiana University Extreme! Lab (<http://www.extreme.indiana.edu/>)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Indiana Univeristy" and "Indiana Univeristy Extreme! Lab" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <http://www.extreme.indiana.edu/>.

5. Products derived from this software may not use "Indiana Univeristy"

name nor may "Indiana Univeristy" appear in their name, without prior written permission of the Indiana University.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====
=====
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from the 'org.apache.commons' to the 'clover.org.apache.commons'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

=====
=====
Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity

exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

File: Core.js

Description:

Provides common utility functions and the Class object used internally by the library.

Also provides the <TreeUtil> object for manipulating JSON tree structures

Some of the Basic utility functions and the Class system are based in the MooTools Framework <<http://mootools.net>>. Copyright (c) 2006-2009 Valerio Proietti, <<http://mad4milk.net/>>. MIT license <<http://mootools.net/license.txt>>.

Author:

Nicolas Garcia Belmonte

Copyright:

Copyright 2008-2009 by Nicolas Garcia Belmonte.

Homepage:

<<http://thejit.org>>

Version:

1.1.2

License:

BSD License

- > Redistribution and use in source and binary forms, with or without
- > modification, are permitted provided that the following conditions are met:
- > * Redistributions of source code must retain the above copyright
- > notice, this list of conditions and the following disclaimer.
- > * Redistributions in binary form must reproduce the above copyright
- > notice, this list of conditions and the following disclaimer in the
- > documentation and/or other materials provided with the distribution.
- > * Neither the name of the organization nor the
- > names of its contributors may be used to endorse or promote products
- > derived from this software without specific prior written permission.
- >
- > THIS SOFTWARE IS PROVIDED BY Nicolas Garcia Belmonte ``AS IS" AND ANY
- > EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
- > WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
- > DISCLAIMED. IN NO EVENT SHALL Nicolas Garcia Belmonte BE LIABLE FOR ANY
- > DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
- > (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
- > LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
- > ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
- > (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
- > SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
- */

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent

to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,

excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any

risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.107 x-sync 20210219-snapshot-036812b2

1.107.1 Available under license :

```
# This source code was written by the Go contributors.  
# The master list of contributors is in the main Go distribution,  
# visible at http://tip.golang.org/CONTRIBUTORS.  
Copyright (c) 2009 The Go Authors. All rights reserved.
```

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
- * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.108 pgv-java-grpc-interceptors 0.6.1

1.108.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.109 golang-mock v1.6.0

1.109.1 Available under license :

```
# This is the official list of people who can contribute (and typically
# have contributed) code to the gomock repository.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#
# The submission process automatically checks to make sure
# that people submitting code are listed in this file (by email address).
#
# Names should be added to this file only after verifying that
# the individual or the individual's organization has agreed to
# the appropriate Contributor License Agreement, found here:
#
#   http://code.google.com/legal/individual-cla-v1.0.html
#   http://code.google.com/legal/corporate-cla-v1.0.html
#
# The agreement for individuals can be filled out on the web.
#
# When adding J Random Contributor's name to this file,
# either J's name or J's organization's name should be
# added to the AUTHORS file, depending on whether the
# individual or corporate CLA was used.

# Names should be added to this file like so:
#   Name <email address>
#
# An entry with two email addresses specifies that the
# first address should be used in the submit logs and
# that the second address should be recognized as the
# same person when interacting with Rietveld.

# Please keep the list sorted.

Aaron Jacobs <jacobsa@google.com> <aaronjjacobs@gmail.com>
Alex Reece <awreece@gmail.com>
David Symonds <dsymonds@golang.org>
Ryan Barrett <ryanb@google.com>
```

```

package empty_interface

//go:generate mockgen -package empty_interface -destination mock.go -source input.go -
copyright_file=mock_copyright_header

type Empty interface{}
// This is a mock copyright header.
//
// Lorem ipsum dolor sit amet, consectetur adipiscing elit,
// sed do eiusmod tempor incididunt ut labore et dolore magna
// aliqua. Velit ut tortor pretium viverra suspendisse potenti.
//

// Code generated by MockGen. DO NOT EDIT.
// Source: input.go

// Package empty_interface is a generated GoMock package.
package empty_interface

import (
    gomock "github.com/golang/mock/gomock"
)

// MockEmpty is a mock of Empty interface.
type MockEmpty struct {
    ctrl    *gomock.Controller
    recorder *MockEmptyMockRecorder
}

// MockEmptyMockRecorder is the mock recorder for MockEmpty.
type MockEmptyMockRecorder struct {
    mock *MockEmpty
}

// NewMockEmpty creates a new mock instance.
func NewMockEmpty(ctrl *gomock.Controller) *MockEmpty {
    mock := &MockEmpty{ctrl: ctrl}
    mock.recorder = &MockEmptyMockRecorder{mock}
    return mock
}

// EXPECT returns an object that allows the caller to indicate expected use.
func (m *MockEmpty) EXPECT() *MockEmptyMockRecorder {
    return m.recorder
}
This is a mock copyright header.

Lorem ipsum dolor sit amet, consectetur adipiscing elit,

```

sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Velit ut tortor pretium viverra suspendisse potenti.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.110 kotlin-stdlib-common 1.3.50

1.110.1 Available under license :

Apache-2.0

1.111 prometheus-procfs v0.7.3

1.111.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only

on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

procfs provides functions to retrieve system, kernel and process metrics from the pseudo-filesystem proc.

Copyright 2014-2015 The Prometheus Authors

This product includes software developed at
SoundCloud Ltd. (<http://soundcloud.com/>).

1.112 pgv-java-parent 0.6.1

1.112.1 Available under license :

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.113 aws-java-sdk-::arns 2.17.52

1.113.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.114 kotlin-scripting-compiler-impl-embeddable 1.3.50

1.114.1 Available under license :

The version of Rhino used in GWT is licensed under a dual license,
Netscape Public License 1.1 / GNU General Public License.
The text of the Netscape Public License
is provided below (<http://website-archive.mozilla.org/www.mozilla.org/mpl/MPL/NPL/1.1/>):

AMENDMENTS

The Netscape Public License Version 1.1 ("NPL") consists of the
Mozilla Public License Version 1.1 with the following Amendments,
including Exhibit A-Netscape Public License. Files identified with
"Exhibit A-Netscape Public License" are governed by the Netscape
Public License Version 1.1.

Additional Terms applicable to the Netscape Public License.

I. Effect.

These additional terms described in this Netscape Public
License -- Amendments shall apply to the Mozilla Communicator
client code and to all Covered Code under this License.

II. "Netscape's Branded Code" means Covered Code that Netscape
distributes and/or permits others to distribute under one or
more trademark(s) which are controlled by Netscape but which
are not licensed for use under this License.

III. Netscape and logo.

This License does not grant any rights to use the trademarks
"Netscape", the "Netscape N and horizon" logo or the
"Netscape lighthouse" logo, "Netcenter", "Gecko", "Java" or
"JavaScript", "Smart Browsing" even if such marks are included
in the Original Code or Modifications.

IV. Inability to Comply Due to Contractual Obligation.

Prior to licensing the Original Code under this License, Netscape has licensed third party code for use in Netscape's Branded Code. To the extent that Netscape is limited contractually from making such third party code available under this License, Netscape may choose to reintegrate such code into Covered Code without being required to distribute such code in Source Code form, even if such code would otherwise be considered "Modifications" under this License.

V. Use of Modifications and Covered Code by Initial Developer.

V.1. In General.

The obligations of Section 3 apply to Netscape, except to the extent specified in this Amendment, Section V.2 and V.3.

V.2. Other Products.

Netscape may include Covered Code in products other than the Netscape's Branded Code which are released by Netscape during the two (2) years following the release date of the Original Code, without such additional products becoming subject to the terms of this License, and may license such additional products on different terms from those contained in this License.

V.3. Alternative Licensing.

Netscape may license the Source Code of Netscape's Branded Code, including Modifications incorporated therein, without such Netscape Branded Code becoming subject to the terms of this License, and may license such Netscape Branded Code on different terms from those contained in this License.

VI. Litigation.

Notwithstanding the limitations of Section 11 above, the provisions regarding litigation in Section 11(a), (b) and (c) of the License shall apply to all disputes relating to this License.

EXHIBIT A-Netscape Public License.

"The contents of this file are subject to the Netscape Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/NPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is Mozilla Communicator client code, released March 31, 1998.

The Initial Developer of the Original Code is Netscape Communications Corporation. Portions created by Netscape are Copyright (C) 1998-1999 Netscape Communications Corporation. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the [____] License), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [____] License and not to allow others to use your version of this file under the NPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [____] License. If you do not delete the provisions above, a recipient may use your version of this file under either the NPL or the [____] License."

MOZILLA PUBLIC LICENSE

Version 1.1

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original

Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements

caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.

The Modifications which You create or to which You contribute are governed by the terms of this License, including without

limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims.

If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such

knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.

If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.

You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.

You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.

Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.

Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.

If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the

Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as Multiple-Licensed. Multiple-Licensed means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is _____.

The Initial Developer of the Original Code is
_____. Portions created by
_____ are Copyright (C) _____
_____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the [____] License), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [____] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [____] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid

anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program).

Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest

your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program

except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed

through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

```
This program is free software; you can redistribute it and/or modify  
it under the terms of the GNU General Public License as published by  
the Free Software Foundation; either version 2 of the License, or  
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,  
but WITHOUT ANY WARRANTY; without even the implied warranty of  
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License along  
with this program; if not, write to the Free Software Foundation, Inc.,  
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author  
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
```

This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

```
=====  
/**  
 * Copyright 2010 Tim Down.  
 *  
 * Licensed under the Apache License, Version 2.0 (the "License");  
 * you may not use this file except in compliance with the License.  
 * You may obtain a copy of the License at  
 *  
 * http://www.apache.org/licenses/LICENSE-2.0  
 *  
 * Unless required by applicable law or agreed to in writing, software  
 * distributed under the License is distributed on an "AS IS" BASIS,  
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
 * See the License for the specific language governing permissions and  
 * limitations under the License.  
 */
```

The Trove library is licensed under the Lesser GNU Public License, which is included with the distribution in a file called trove_license.txt.

The PrimeFinder and HashFunctions classes in Trove are subject to the following license restrictions:

Copyright (c) 1999 CERN - European Organization for Nuclear Research.

Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. CERN makes no representations about the suitability of this software for any purpose. It is provided "as is" without expressed or implied warranty.

Copyright (c) 2005-2010 Sam Stephenson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

/*

* Copyright 2010-2018 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

The Apache 2 license (given in full in LICENSE.txt) applies to all code in this repository which is copyright by JetBrains. The following sections of the repository contain third-party code, to which different licenses may apply:

Kotlin Compiler

The following modules contain third-party code and are incorporated into the Kotlin compiler and/or the Kotlin IntelliJ IDEA plugin:

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/inline/MaxStackSizeAndLocalsCalculator.java

- License: BSD ([license/third_party/asm_license.txt][asm])
- Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/inline/MaxLocalsCalculator.java
- License: BSD ([license/third_party/asm_license.txt][asm])
- Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/optimization/common/MethodAnalyzer.kt
- License: BSD ([license/third_party/asm_license.txt][asm])
- Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: core/reflection.jvm/src/kotlin.reflect/jvm/internal/pcollections
- License: MIT ([license/third_party/pcollections_LICENSE.txt][pcollections])
- Origin: Derived from PCollections, A Persistent Java Collections Library (<https://pcollections.org/>)

- Path: eval4j/src/org/jetbrains/eval4j/interpreterLoop.kt
- License: BSD ([license/third_party/asm_license.txt][asm])
- Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/optimization/common/OptimizationBasicInterpreter.java
- License: BSD ([license/third_party/asm_license.txt][asm])
- Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: js/js.ast
- License: BSD ([license/third_party/dart_LICENSE.txt][dart])
- Origin: Originally part of the Dart compiler, (c) 2011 the Dart Project Authors,

- Path: js/js.inliner/src/org/jetbrains/kotlin/js/inline/FunctionInlineMutator.kt
- License: BSD ([license/third_party/dart_LICENSE.txt][dart])
- Origin: Originally part of the Dart compiler, (c) 2011 the Dart Project Authors,

- Path: js/js.parser/src/com/google
- License: Netscape Public License 1.1 ([license/third_party/rhino_LICENSE.txt][rhino])
- Origin: Originally part of GWT, (C) 2007-08 Google Inc., distributed under the Apache 2 license. The code is derived from Rhino, (C) 1997-1999 Netscape Communications Corporation, distributed under the Netscape Public License.

- Path: js/js.translator/qunit/qunit.js
- License: MIT ([license/third_party/qunit_license.txt][qunit])
- Origin: QUnit, Copyright (c) 2012 John Resig, Jrn Zaefferer,

- Path: libraries/stdlib/src/kotlin/collections
- License: Apache 2 ([license/third_party/gwt_license.txt][gwt])

- Origin: Derived from GWT, (C) 2007-08 Google Inc.
- Path: libraries/stdlib/unsigned/src/kotlin/UnsignedUtils.kt
 - License: Apache 2 ([license/third_party/guava_license.txt][guava])
 - Origin: Derived from Guava's UnsignedLongs, (C) 2011 The Guava Authors
- Path: libraries/stdlib/jvm/src/kotlin/util/MathJVM.kt
 - License: Boost Software License 1.0 ([license/third_party/boost_LICENSE.txt][boost])
 - Origin: Derived from boost special math functions, Copyright Eric Ford & Hubert Holin 2001.
- Path: libraries/stdlib/js/src/kotlin/collections
 - License: Apache 2 ([license/third_party/gwt_license.txt][gwt])
 - Origin: Derived from GWT, (C) 2007-08 Google Inc.
- Path: libraries/stdlib/js-v1/src/js/long.js
 - License: Apache 2 ([license/third_party/closure-compiler_LICENSE.txt][closure-compiler])
 - Origin: Google Closure Library, Copyright 2009 The Closure Library Authors
- Path: libraries/stdlib/js-v1/src/js/polyfills.js
 - License: Boost Software License 1.0 ([license/third_party/boost_LICENSE.txt][boost])
 - Origin: Derived from boost special math functions, Copyright Eric Ford & Hubert Holin 2001.
- Path: plugins/lint/android-annotations
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project
- Path: plugins/lint/lint-api
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project
- Path: plugins/lint/lint-checks
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project
- Path: plugins/lint/lint-idea
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

Kotlin Test Data

The following source code is used for testing the Kotlin compiler and/or plugin and is not incorporated into any distributions of the compiler, libraries or plugin:

- Path: third-party/annotations/android
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project
- Path: third-party/annotations/com/android

- License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
- Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: third-party/annotations/org/eclipse
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])
 - Origin: Eclipse JDT, Copyright (c) 2011, 2013 Stephan Herrmann and others.

- Path: third-party/annotations/androidx
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: third-party/annotations/edu/umd/cs/findbugs
 - License: LGPL 2.1 ([license/third_party/testdata/findbugs_license.txt][findbugs])
 - Origin: Bytecode Analysis Framework, Copyright (C) 2005 University of Maryland

- Path: third-party/jdk8-annotations/org/eclipse
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])
 - Origin: Eclipse JDT, Copyright (c) 2011, 2013 Stephan Herrmann and others.

- Path: third-party/annotations/io/reactivex
 - License: Apache 2 ([license/third_party/testdata/rxjava_license.txt][rxjava])
 - Origin: RxJava, Copyright (c) 2016-present, RxJava Contributors

- Path: third-party/annotations/lombok
 - License: MIT ([license/third_party/testdata/lombok_license.txt][lombok])
 - Origin: Project Lombok, Copyright (C) 2009-2013 The Project Lombok Authors

- Path: idea/idea-android/tests/org/jetbrains/kotlin/android/AndroidTestBase.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: idea/testData/android/lintQuickfix/requiresApi/RequiresApi.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: idea/testData/android/lint/IntRange.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: idea/testData/android/lint/RequiresPermission.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: libraries/tools/kotlin-gradle-plugin-integration-
tests/src/test/resources/testProject/allOpenSpring/src/org/springframework/stereotype/Component.java
 - License: Apache 2 ([license/third_party/testdata/spring_license.txt][spring])
 - Origin: Spring Framework, Copyright 2002-2007 the original author or authors.

- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/resources/testProject/AndroidDaggerProject
 - License: Apache 2 ([license/third_party/testdata/dagger_license.txt][dagger])
 - Origin: Dagger, Copyright (C) 2013 Square, Inc.
- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/resources/testProject/kapt2
 - License: Apache 2 ([license/third_party/testdata/dagger_license.txt][dagger])
 - Origin: Dagger, Copyright (C) 2013 Square, Inc.
- Path: libraries/tools/kotlin-maven-plugin-test/src/it/test-allopen-spring/src/main/java/org/springframework/stereotype/Component.java
 - License: Apache 2 ([license/third_party/testdata/spring_license.txt][spring])
 - Origin: Spring Framework, Copyright 2002-2007 the original author or authors.

Kotlin Tools and Libraries Tests

The following source code is used for testing the Kotlin tools and/or libraries and is not incorporated into any distributions of the tools or libraries:

- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/kotlin/org/jetbrains/kotlin/gradle/PluginsDslIT.kt
 - License: Apache 2 ([license/third_party/testdata/gradle_license.txt][gradle])
 - Origin: Gradle, Copyright 2002-2017 Gradle, Inc.
- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/target/test-classes/testProject/noArgJpa/src/javax/persistence/Entity.java
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse]) and Eclipse Distribution License - v1.0 ([license/third_party/testdata/eclipse_distribution_license.txt][eclipse-distribution])
 - Origin: javax.persistence, Copyright (c) 2008, 2017 Sun Microsystems, Oracle Corporation.
- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/resources/testProject/noArgJpa/src/javax/persistence/Entity.java
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse]) and Eclipse Distribution License - v1.0 ([license/third_party/testdata/eclipse_distribution_license.txt][eclipse-distribution])
 - Origin: javax.persistence, Copyright (c) 2008, 2017 Sun Microsystems, Oracle Corporation.
- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/resources/testProject/noArgJpa/src/javax/persistence/Embeddable.java
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse]) and Eclipse Distribution License - v1.0 ([license/third_party/testdata/eclipse_distribution_license.txt][eclipse-distribution])
 - Origin: javax.persistence, Copyright (c) 2008, 2017 Sun Microsystems, Oracle Corporation.

Example Code

The following code is provided as examples and is not incorporated into any distributions of the compiler, libraries or plugin:

- Path: libraries/examples/browser-example/src/js/jquery.js
 - License: MIT ([license/third_party/jquery_license.txt][jquery])
 - Origin: jQuery JavaScript Library v1.6.2, Copyright 2011, John Resig

- Path: libraries/examples/browser-example-with-library/src/js/jquery.js
 - License: MIT ([license/third_party/jquery_license.txt][jquery])
 - Origin: jQuery JavaScript Library v1.6.2, Copyright 2011, John Resig

[aosp]: third_party/aosp_license.txt
 [asm]: third_party/asm_license.txt
 [boost]: third_party/boost_LICENSE.txt
 [closure-compiler]: third_party/closure-compiler_LICENSE.txt
 [dagger]: third_party/testdata/dagger_license.txt
 [dart]: third_party/dart_LICENSE.txt
 [eclipse]: third_party/testdata/eclipse_license.txt
 [eclipse-distribution]: third_party/testdata/eclipse_distribution_license.txt
 [findbugs]: third_party/testdata/findbugs_license.txt
 [gradle]: third_party/testdata/gradle_license.txt
 [guava]: third_party/guava_license.txt
 [gwt]: third_party/gwt_license.txt
 [jquery]: third_party/jquery_license.txt
 [lombok]: third_party/testdata/lombok_license.txt
 [pcollections]: third_party/pcollections_LICENSE.txt
 [qunit]: third_party/qunit_license.txt
 [rhino]: third_party/rhino_LICENSE.txt
 [rxjava]: third_party/testdata/rxjava_license.txt
 [spring]: third_party/testdata/spring_license.txt
 Copyright (C) 2009-2015 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

SCALA LICENSE

Copyright (c) 2002-2012 EPFL, Lausanne, unless otherwise specified.
All rights reserved.

This software was developed by the Programming Methods Laboratory of the
Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland.

Permission to use, copy, modify, and distribute this software in source
or binary form for any purpose with or without fee is hereby granted,
provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the EPFL nor the names of its contributors
may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

/* PRESENT 1 */

/* PRESENT 2 */

// PRESENT 3

/** ABSENT */

package/* ABSENT 1 */ normal

/* ABSENT 2 */

// COMMENTS: 3

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2005 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions

are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Eclipse Distribution License - v 1.0

Copyright (c) 2007, Eclipse Foundation, Inc. and its licensors.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/* ABSENT */

class Some

// COMMENTS: 0

This license applies to all parts of Dart that are not externally maintained libraries. The external maintained libraries used by Dart are:

7-Zip - in third_party/7zip

JSCORE - in runtime/third_party/jscre

Ant - in third_party/apache_ant

args4j - in third_party/args4j

bzip2 - in third_party/bzip2

dromaeo - in samples/third_party/dromaeo

Eclipse - in third_party/eclipse

gsutil = in third_party/goutil

Guava - in third_party/guava

hamcrest - in third_party/hamcrest

HttpLib2 - in samples/third_party/httpLib2

JSON - in third_party/json

JUnit - in third_party/junit

Oauth - in samples/third_party/oauth2client

Rhino - in third_party/rhino

weberknecht - in third_party/weberknecht

The libraries may have their own licenses; we recommend you read them, as their terms may differ from the terms below.

Copyright 2012, the Dart project authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

*/

```
package org.jetbrains.idl2k.util
```

```
import org.xml.sax.InputSource
```

```
import java.io.File
```

```
import javax.xml.xpath.XPathFactory
```

```
fun readCopyrightNoticeFromProfile(copyrightProfile: File): String {
```

```
    val template = copyrightProfile.reader().use { reader ->
```

```
        XPathFactory.newInstance().newXPath().evaluate("/component/copyright/option[@name='notice']/@value",  
InputSource(reader))
```

```
    }
```

```
    val yearTemplate = "&#36;today.year"
```

```
    val year = java.time.LocalDate.now().year.toString()
```

```
    assert(yearTemplate in template)
```

```
    return template.replace(yearTemplate, year).lines().joinToString("", prefix = "/*\n", postfix = " *\n") { " * $it\n" }
```

```
/* PRESENT */
```

```
fun some() {}
```

```
// COMMENTS: 1
```

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or

documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,

work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====
Gradle Subcomponents:

License for the slf4j package

SLF4J License

Copyright (c) 2004-2007 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

These terms are identical to those of the MIT License, also called the X License or the X11 License, which is a simple, permissive non-copyleft free software license. It is deemed compatible with virtually all types of licenses, commercial or otherwise. In particular, the Free Software Foundation has declared it compatible with GNU GPL. It is also known to be approved by the Apache Software Foundation as compatible with Apache Software License.

License for the JUnit package

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the

rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL,

SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not

expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

License for the JCIFS package

JCIFS License

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid

distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less

of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a

portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or

table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or

derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse

engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on

which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with

this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and

"any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!

/* PRESENT */

package normal

// COMMENTS: 1

Google Dart Js backend was removed - <https://code.google.com/p/dart/source/detail?r=4771>

According to <http://www.apache.org/legal/3party.html> we can include "Google Dart Js backend" in source form, because code license is "New BSD License" (Authorized License).

This part of code will be removed when kotlin will be rewritten on kotlin.

Copyright JS Foundation and other contributors, <https://js.foundation/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at <https://github.com/jquery/jquery>

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

```
/* PRESENT */
```

```
foo()
```

```
// COMMENTS: 1
```

```
Boost Software License - Version 1.0 - August 17th, 2003
```

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by

a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```
/*
```

```
* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
```

```
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
```

```
*/
```

```
package org.jetbrains.kotlin.copyright
```

```
import junit.framework.AssertionFailedError
```

```
import org.jetbrains.kotlin.idea.copyright.UpdateKotlinCopyright
```

```
import org.jetbrains.kotlin.idea.test.KotlinLightCodeInsightFixtureTestCase
```

```
import org.jetbrains.kotlin.idea.test.PluginTestCaseBase
```

```
import org.jetbrains.kotlin.test.InTextDirectivesUtils
```

```
import org.junit.Assert
```

```
import java.io.File
```

```
abstract class AbstractUpdateKotlinCopyrightTest : KotlinLightCodeInsightFixtureTestCase() {
```

```
    fun doTest(path: String) {
```

```
        myFixture.configureByFile(path)
```

```
        val fileText = myFixture.file.text.trim()
```

```
        val expectedNumberOfComments = InTextDirectivesUtils.getPrefixedInt(fileText, "// COMMENTS: ") ?: run {
```

```
            if (fileText.isNotEmpty()) {
```

```
                throw AssertionError("Every test should assert number of comments with `COMMENTS`
```

```
directive")
```

```
            } else {
```

```
                0
```

```
            }
```

```
        }
```

```
        val comments = UpdateKotlinCopyright.getExistentComments(myFixture.file)
```

```
        for (comment in comments) {
```

```
            val commentText = comment.text
```

```
            when {
```

```
                commentText.contains("PRESENT") -> {
```

```
                }
```

```
                commentText.contains("ABSENT") -> {
```

```
                    throw AssertionError("Unexpected comment found: `$commentText`")
```

```
                }
```

```
            } else -> {
```

```

        throw AssertionError("A comment with bad directive found: `${commentText}`")
    }
}

Assert.assertEquals(
    "Wrong number of comments found:\n${comments.joinToString(separator = "\n") { it.text }}\n",
    expectedNumberOfComments, comments.size
)
}

override fun getTestDataPath() = File(PluginTestCaseBase.getTestDataPathBase(), "/copyright").path +
File.separator
}
/*
* Copyright 2000-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
*/

package tasks

import groovy.util.Node
import groovy.util.XmlParser
import org.gradle.api.DefaultTask
import org.gradle.api.Project
import org.gradle.api.tasks.Input
import org.gradle.api.tasks.InputFile
import org.gradle.api.tasks.OutputFile
import org.gradle.api.tasks.TaskAction
import java.io.File
import java.util.*

open class WriteCopyrightToFile : DefaultTask() {

    @InputFile
    var path = project.file("${project.rootDir}/.idea/copyright/apache.xml")

    @OutputFile
    var outputFile: File? = null

    @Input
    var commented: Boolean = true

    @TaskAction
    fun write() {

```

```

if (commented) {
    outputFile!!.writeText(project.readCopyrightCommented())
} else {
    outputFile!!.writeText(project.readCopyright())
}
}

fun Project.readCopyright(): String {
    val file = rootDir.resolve(".idea/copyright/apache.xml")

    assert(file.exists()) {
        "File $file with copyright not found"
    }

    val xmlParser = XmlParser()
    val node = xmlParser.parse(file)
    assert(node.attribute("name") == "CopyrightManager") {
        "Format changed occasionally?"
    }

    val copyrightBlock = node.children().filterIsInstance<Node>().single()
    val noticeNode = copyrightBlock.children().filterIsInstance<Node>().single { it.attribute("name") == "notice" }
    return noticeNode.attribute("value").toString().replace("&#36;today.year",
GregorianCalendar()[Calendar.YEAR].toString())
}

fun Project.readCopyrightCommented(): String {
    return "/*\n" + readCopyright().prependIndent(" * ") + "\n */"
}
}

```

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public

Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must

be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated

```
/*  
* Copyright 2010-2017 JetBrains s.r.o.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from

such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in

themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the

Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under

Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on

the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to

refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing

and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
The files in this package are taken from mozilla's Rhino project.
See <http://www.mozilla.org/rhino/>

The files modified from Rhino 1.5R3
(<ftp://ftp.mozilla.org/pub/js/rhino15R3.zip>).

--

The contents of this package are subject to the Netscape Public
License Version 1.1 (the "License"); you may not use this file
except in compliance with the License. You may obtain a copy of
the License at <http://www.mozilla.org/NPL/>

Software distributed under the License is distributed on an "AS
IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or
implied. See the License for the specific language governing
rights and limitations under the License.

The Original Code is Rhino code, released
May 6, 1999.

The Initial Developer of the Original Code is Netscape
Communications Corporation. Portions created by Netscape are
Copyright (C) 1997-2000 Netscape Communications Corporation. All
Rights Reserved.

Alternatively, the contents of this file may be used under the terms of the GNU Public License (the "GPL"), in which case the provisions of the GPL are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the GPL and not to allow others to use your version of this file under the NPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the GPL. If you do not delete the provisions above, a recipient may use your version of this file under either the NPL or the GPL.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work

(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses

granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"

replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2000-2018 JetBrains s.r.o.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

```
=====
== NOTICE file corresponding to the section 4 d of      ==
== the Apache License, Version 2.0,                    ==
== in this case for the Kotlin Compiler distribution.    ==
=====
```

Kotlin Compiler

Copyright 2010-2019 JetBrains s.r.o and respective authors and developers

Copyright (c) 2008 Harold Cooper

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

- a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
- b) in the case of each subsequent Contributor:
 - i) changes to the Program, and
 - ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

- a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
- b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under

Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution.

No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

- a) it complies with the terms and conditions of this Agreement; and
- b) its license agreement:
 - i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 - ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 - iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

- a) it must be made available under this Agreement; and
- b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

- a) promptly notify the Commercial Contributor in writing of such claim, and
- b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result,

the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The MIT License

Copyright (c) 2003, Kohsuke Kawaguchi

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright JS Foundation and other contributors, <https://js.foundation>

This software consists of voluntary contributions made by many

individuals. For exact contribution history, see the revision history available at <https://github.com/qunitjs/qunit>

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some

specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and

is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1

above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do

this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline

functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more

than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING

RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.115 cocood-freecache v1.2.0

1.115.1 Available under license :

The MIT License

Copyright (c) 2015 Ewan Chou.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.116 JetBrains-Kotlin-Kotlin-Scripting- Compiler-Embeddable 1.3.50

1.116.1 Available under license :

Apache License
Version 2.0, January 2004

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally

submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or

implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.117 goconvey v1.7.2

1.117.1 Available under license :

Copyright (c) 2016 SmartyStreets, LLC

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

NOTE: Various optional and subordinate components carry their own licensing requirements and restrictions. Use of those components is subject to the terms and conditions outlined the respective license of each component.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.118 slf4j-api-module 1.7.32

1.118.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/**
 * Copyright (c) 2004-2011 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
 * LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
 * OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */
```

Found in path(s):

```
* /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/MDC.java
* /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/Logger.java
* /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-
jar/org/slf4j/impl/StaticMDCBinder.java
* /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-
jar/org/slf4j/spi/MarkerFactoryBinder.java
* /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-
jar/org/slf4j/helpers/BasicMarkerFactory.java
* /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-
jar/org/slf4j/helpers/SubstituteLogger.java
* /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/LoggerFactory.java
* /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-
jar/org/slf4j/helpers/BasicMDCAdapter.java
* /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/MarkerFactory.java
* /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-
jar/org/slf4j/helpers/SubstituteLoggerFactory.java
```

- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/Marker.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/helpers/BasicMarker.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/helpers/MarkerIgnoringBase.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/helpers/NOPLogger.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/spi/LocationAwareLogger.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/helpers/NOPMDCAadapter.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/helpers/NOPLoggerFactory.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/impl/StaticMarkerBinder.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/impl/StaticLoggerBinder.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/spi/LoggerFactoryBinder.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/IMarkerFactory.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/ILoggerFactory.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/helpers/Util.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/helpers/MessageFormatter.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/spi/MDCAdapter.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/helpers/NamedLoggerBase.java
- * /opt/cola/permits/1267277703_1648511906.32/0/slf4j-api-1-7-32-sources-jar/org/slf4j/helpers/FormattingTuple.java

1.119 netty/resolver/dns/classes/macos

4.1.72.Final

1.119.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
```

* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1411867064_1662690034.0692346/0/netty-resolver-dns-classes-macos-4-1-72-final-sources-2-jar/io/netty/resolver/dns/macos/DnsResolver.java
* /opt/cola/permits/1411867064_1662690034.0692346/0/netty-resolver-dns-classes-macos-4-1-72-final-sources-2-jar/io/netty/resolver/dns/macos/package-info.java
* /opt/cola/permits/1411867064_1662690034.0692346/0/netty-resolver-dns-classes-macos-4-1-72-final-sources-2-jar/io/netty/resolver/dns/macos/MacOSDnsServerAddressStreamProvider.java

No license file was found, but licenses were detected in source scan.

<!--

~ Copyright 2021 The Netty Project

~

~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:

~

~ <https://www.apache.org/licenses/LICENSE-2.0>

~

~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

-->

Found in path(s):

* /opt/cola/permits/1411867064_1662690034.0692346/0/netty-resolver-dns-classes-macos-4-1-72-final-sources-2-jar/META-INF/maven/io.netty/netty-resolver-dns-classes-macos/pom.xml

1.120 go-humanize v1.0.0

1.120.1 Available under license :

Copyright (c) 2005-2008 Dustin Sallings <dustin@spy.net>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

<<http://www.opensource.org/licenses/mit-license.php>>

1.121 jackson-databind 2.13.0

1.121.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the

Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.122 yaml-for-go 20200512-snapshot-9f266ea9

1.122.1 Available under license :

This project is covered by two different licenses: MIT and Apache.

MIT License

The following files were ported to Go from C files of libyaml, and thus are still covered by their original MIT license, with the additional copyright starting in 2011 when the project was ported over:

apic.go emitterc.go parserc.go readerc.go scannerc.go
writerc.go yamlh.go yamlprivateh.go

Copyright (c) 2006-2010 Kirill Simonov

Copyright (c) 2006-2011 Kirill Simonov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License

All the remaining project files are covered by the Apache license:

Copyright (c) 2011-2019 Canonical Ltd

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Copyright 2011-2016 Canonical Ltd.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.123 aws-sdk-for-java-core 1.12.26

1.123.1 Available under license :

Apache License

Version 2.0, January 2004

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

“License” shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

“Licensor” shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

“Legal Entity” shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, “control” means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

“You” (or “Your”) shall mean an individual or Legal Entity exercising permissions granted by this License.

“Source” form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

“Object” form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

“Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

“Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

“Contribution” shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, “submitted” means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as “Not a Contribution.”

“Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a “NOTICE” text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.124 go-spew v1.1.1

1.124.1 Available under license :

ISC License

Copyright (c) 2012-2016 Dave Collins <dave@davec.name>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR

ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.125 netty-codec-smtp 4.1.72.Final

1.125.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-
jar/io/netty/handler/codec/smtp/SmtpRequestEncoder.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-
jar/io/netty/handler/codec/smtp/LastSmtpContent.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-
jar/io/netty/handler/codec/smtp/DefaultSmtpRequest.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-
jar/io/netty/handler/codec/smtp/SmtpCommand.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-
jar/io/netty/handler/codec/smtp/SmtpResponseDecoder.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-
jar/io/netty/handler/codec/smtp/SmtpResponse.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-
jar/io/netty/handler/codec/smtp/DefaultSmtpResponse.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-
jar/io/netty/handler/codec/smtp/DefaultSmtpContent.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-
jar/io/netty/handler/codec/smtp/SmtpContent.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-
jar/io/netty/handler/codec/smtp/SmtpRequest.java
```

```
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-jar/io/netty/handler/codec/smtp/SmtRequests.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-jar/io/netty/handler/codec/smtp/DefaultLastSmtpContent.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-jar/io/netty/handler/codec/smtp/package-info.java
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-jar/io/netty/handler/codec/smtp/SmtUtils.java
```

No license file was found, but licenses were detected in source scan.

<!--

~ Copyright 2016 The Netty Project

~

~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:

~

~ <https://www.apache.org/licenses/LICENSE-2.0>

~

~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

-->

Found in path(s):

```
* /opt/cola/permits/1411866710_1662683897.8805194/0/netty-codec-smtp-4-1-72-final-sources-1-jar/META-INF/maven/io.netty/netty-codec-smtp/pom.xml
```

1.126 aws-java-sdk-::-third-party-::-jackson-core 2.17.52

1.126.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed

with the accompanying software if such terms are included in the directory containing the accompanying software.
Such other license terms will then apply in lieu of the terms of the software license above.

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0
To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by
Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

Apache License
Version 2.0, January 2004

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally

submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or

implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.127 pmezard-go-difflib 1.0.0

1.127.1 Available under license :

Copyright (c) 2013, Patrick Mezard
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.128 error_prone_annotations 2.7.1

1.128.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2014 The Error Prone Authors.

*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/NoAllocation.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/LockMethod.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/UnlockMethod.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The Error Prone Authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/OverridingMethodsMustInvokeSuper.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/DoNotCall.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/GuardedBy.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/CheckReturnValue.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2016 The Error Prone Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
 * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-
jar/com/google/errorprone/annotations/FormatString.java
 * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-
jar/com/google/errorprone/annotations/CompatibleWith.java
 * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-
jar/com/google/errorprone/annotations/RestrictedApi.java
 * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-
jar/com/google/errorprone/annotations/MustBeClosed.java
 * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-
jar/com/google/errorprone/annotations/DoNotMock.java
 * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-
jar/com/google/errorprone/annotations/FormatMethod.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2015 The Error Prone Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/Var.java
- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/LazyInit.java
- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/IncompatibleModifiers.java
- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/RequiredModifiers.java
- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/Immutable.java
- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/CompileTimeConstant.java
- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/SuppressPackageLocation.java
- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/ForOverride.java
- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/CanIgnoreReturnValue.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2021 The Error Prone Authors.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/InlineMeValidationDisabled.java
- * /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/InlineMe.java

1.129 apache-commons-lang 3.8.1

1.129.1 Available under license :

Apache Commons Lang
Copyright 2001-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the

editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the

Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the

same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.130 kubernetes-apimachinery 20191123-snapshot-4c4803ed

1.130.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.131 duct-tape 1.0.8

1.131.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Richard North

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.132 grpc-netty-shaded 1.39.0

1.132.1 Available under license :

LICENSE ISSUES

=====

The OpenSSL toolkit stays under a double license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts.

OpenSSL License

/* =====

- * Copyright (c) 1998-2019 The OpenSSL Project. All rights reserved.
- *
- * Redistribution and use in source and binary forms, with or without
- * modification, are permitted provided that the following conditions
- * are met:
- *
- * 1. Redistributions of source code must retain the above copyright
- * notice, this list of conditions and the following disclaimer.
- *
- * 2. Redistributions in binary form must reproduce the above copyright
- * notice, this list of conditions and the following disclaimer in
- * the documentation and/or other materials provided with the
- * distribution.
- *
- * 3. All advertising materials mentioning features or use of this
- * software must display the following acknowledgment:
- * "This product includes software developed by the OpenSSL Project
- * for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"
- *
- * 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to

* endorse or promote products derived from this software without
 * prior written permission. For written permission, please contact
 * openssl-core@openssl.org.
 *
 * 5. Products derived from this software may not be called "OpenSSL"
 * nor may "OpenSSL" appear in their names without prior written
 * permission of the OpenSSL Project.
 *
 * 6. Redistributions of any form whatsoever must retain the following
 * acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"
 *
 * THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS" AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 * =====
 *
 * This product includes cryptographic software written by Eric Young
 * (eay@cryptsoft.com). This product includes software written by Tim
 * Hudson (tjh@cryptsoft.com).
 *
 */

Original SSLeay License

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to. The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * lhash, DES, etc., code; not just the SSL code. The SSL documentation
 * included with this distribution is covered by the same copyright terms

- * except that the holder is Tim Hudson (tjh@cryptsoft.com).
- *
- * Copyright remains Eric Young's, and as such any Copyright notices in
- * the code are not to be removed.
- * If this package is used in a product, Eric Young should be given attribution
- * as the author of the parts of the library used.
- * This can be in the form of a textual message at program startup or
- * in documentation (online or textual) provided with the package.
- *
- * Redistribution and use in source and binary forms, with or without
- * modification, are permitted provided that the following conditions
- * are met:
- * 1. Redistributions of source code must retain the copyright
- * notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright
- * notice, this list of conditions and the following disclaimer in the
- * documentation and/or other materials provided with the distribution.
- * 3. All advertising materials mentioning features or use of this software
- * must display the following acknowledgement:
- * "This product includes cryptographic software written by
- * Eric Young (eay@cryptsoft.com)"
- * The word 'cryptographic' can be left out if the rouines from the library
- * being used are not cryptographic related :-).
- * 4. If you include any Windows specific code (or a derivative thereof) from
- * the apps directory (application code) you must include an acknowledgement:
- * "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
- *
- * THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND
- * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
- * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
- * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
- * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
- * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
- * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
- * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
- * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
- * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
- * SUCH DAMAGE.
- *
- * The licence and distribution terms for any publically available version or
- * derivative of this code cannot be changed. i.e. this code cannot simply be
- * copied and put under another distribution licence
- * [including the GNU Public Licence.]
- */

BoringSSL is a fork of OpenSSL. As such, large parts of it fall under OpenSSL licensing. Files that are completely new have a Google copyright and an ISC license. This license is reproduced at the bottom of this file.

Contributors to BoringSSL are required to follow the CLA rules for Chromium:

<https://cla.developers.google.com/cla>

Files in `third_party/` have their own licenses, as described therein. The MIT license, for `third_party/fiat`, which, unlike other `third_party` directories, is compiled into non-test libraries, is included below.

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

The following are Google-internal bug numbers where explicit permission from some authors is recorded for use of their work. (This is purely for our own record keeping.)

27287199

27287880

27287883

OpenSSL License

```
/* =====
```

```
* Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.
```

```
*
```

```
* Redistribution and use in source and binary forms, with or without  
* modification, are permitted provided that the following conditions  
* are met:
```

```
*
```

```
* 1. Redistributions of source code must retain the above copyright  
* notice, this list of conditions and the following disclaimer.
```

```
*
```

```
* 2. Redistributions in binary form must reproduce the above copyright  
* notice, this list of conditions and the following disclaimer in  
* the documentation and/or other materials provided with the  
* distribution.
```

```
*
```

```
* 3. All advertising materials mentioning features or use of this  
* software must display the following acknowledgment:  
* "This product includes software developed by the OpenSSL Project  
* for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
```

```
*
```

```
* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to  
* endorse or promote products derived from this software without  
* prior written permission. For written permission, please contact  
* openssl-core@openssl.org.
```

```
*
```

```
* 5. Products derived from this software may not be called "OpenSSL"  
* nor may "OpenSSL" appear in their names without prior written  
* permission of the OpenSSL Project.
```

```
*
```

```
* 6. Redistributions of any form whatsoever must retain the following  
* acknowledgment:
```

* "This product includes software developed by the OpenSSL Project
* for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS" AND ANY
* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.

* =====

*
* This product includes cryptographic software written by Eric Young
* (eay@cryptsoft.com). This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).

*
*/

Original SSLeay License

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
* All rights reserved.
*
* This package is an SSL implementation written
* by Eric Young (eay@cryptsoft.com).
* The implementation was written so as to conform with Netscapes SSL.
*
* This library is free for commercial and non-commercial use as long as
* the following conditions are aheared to. The following conditions
* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code. The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
*
* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:

* 1. Redistributions of source code must retain the copyright
 * notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.

* 3. All advertising materials mentioning features or use of this software
 * must display the following acknowledgement:
 * "This product includes cryptographic software written by
 * Eric Young (eay@cryptsoft.com)"
 * The word 'cryptographic' can be left out if the routines from the library
 * being used are not cryptographic related :-).

* 4. If you include any Windows specific code (or a derivative thereof) from
 * the apps directory (application code) you must include an acknowledgement:
 * "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
 *

* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *

* The licence and distribution terms for any publically available version or
 * derivative of this code cannot be changed. i.e. this code cannot simply be
 * copied and put under another distribution licence
 * [including the GNU Public Licence.]
 */

ISC license used for completely new code in BoringSSL:
 /* Copyright (c) 2015, Google Inc.
 *
 * Permission to use, copy, modify, and/or distribute this software for any
 * purpose with or without fee is hereby granted, provided that the above
 * copyright notice and this permission notice appear in all copies.
 *
 * THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
 * WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
 * MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY
 * SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
 * WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
 * OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
 * CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. */

The code in third_party/ fiat carries the MIT license:
 Copyright (c) 2015-2016 the fiat-crypto authors (see

<https://github.com/mit-plv/ fiat-crypto/blob/master/AUTHORS>).

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Licenses for support code

Parts of the TLS test suite are under the Go license. This code is not included in BoringSSL (i.e. libcrypto and libssl) when compiled, however, so distributing code linked against BoringSSL does not trigger this license: Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

BoringSSL uses the Chromium test infrastructure to run a continuous build, trybots etc. The scripts which manage this, and the script for generating build metadata, are under the Chromium license. Distributing code linked against

BoringSSL does not trigger this license.

Copyright 2015 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity

exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The Netty Project

=====

Please visit the Netty web site for more information:

* <http://netty.io/>

Copyright 2016 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product contains a forked and modified version of Tomcat Native

- * LICENSE:
 - * license/LICENSE.tomcat-native.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <http://tomcat.apache.org/native-doc/>
 - * <https://svn.apache.org/repos/asf/tomcat/native/>

This product contains the Maven wrapper scripts from 'Maven Wrapper', that provides an easy way to ensure a user has everything necessary to run the Maven build.

- * LICENSE:
 - * license/LICENSE.mvn-wrapper.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/takari/maven-wrapper>

This product contains small piece of code to support AIX, taken from netbsd.

- * LICENSE:
 - * license/LICENSE.aix-netbsd.txt (OpenSSL License)
- * HOMEPAGE:
 - * <https://ftp.netbsd.org/pub/NetBSD/NetBSD-current/src/crypto/external/bsd/openssl/dist>

This product contains code from boringssl.

- * LICENSE (Combination ISC and OpenSSL license)
 - * license/LICENSE.boringssl.txt (Combination ISC and OpenSSL license)
- * HOMEPAGE:
 - * <https://boringssl.googlesource.com/boringssl/>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.133 google-api-grpc-proto-google-common-protos 2.0.1

1.133.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// Licensed under the Apache License, Version 2.0 (the "License");  
// you may not use this file except in compliance with the License.  
// You may obtain a copy of the License at  
// http://www.apache.org/licenses/LICENSE-2.0  
// distributed under the License is distributed on an "AS IS" BASIS,  
// agreement (which includes confidentiality provisions). These features may
```

Found in path(s):

```
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/api/launch_stage.proto
```

No license file was found, but licenses were detected in source scan.

```
// Licensed under the Apache License, Version 2.0 (the "License");  
// you may not use this file except in compliance with the License.  
// You may obtain a copy of the License at  
// http://www.apache.org/licenses/LICENSE-2.0  
// distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):

```
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/api/label.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/type/postal_address.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/type/datetime.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/api/source_info.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/type/quaternion.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/api/field_behavior.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/type/latlng.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/api/httpbody.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/type/money.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/logging/type/http_request.proto
```

* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/type/calendar_period.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/rpc/code.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/geo/type/viewport.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/client.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/longrunning/operations.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/billing.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/backend.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/consumer.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/type/fraction.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/system_parameter.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/rpc/status.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/documentation.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/http.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/rpc/context/attribute_context.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/quota.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/type/date.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/logging.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/cloud/audit/audit_log.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/distribution.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/config_change.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/metric.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/type/dayofweek.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/endpoint.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/usage.proto

* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/auth.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/type/expr.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/log.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/annotations.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/service.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/monitoring.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/resource.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/logging/type/log_severity.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/type/color.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/rpc/error_details.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/context.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/type/timeofday.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/control.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-jar/google/api/monitored_resource.proto

1.134 prometheus-client-model v0.2.0

1.134.1 Available under license :

Data model artifacts for Prometheus.

Copyright 2012-2015 The Prometheus Authors

This product includes software developed at
SoundCloud Ltd. (<http://soundcloud.com/>).

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but

excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its

distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,

unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.135 aws-java-sdk-aws-core 2.17.52

1.135.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade

names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier

identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and

limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed

with the accompanying software if such terms are included in the directory containing the accompanying software.

Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by

Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:

Apache Commons Lang

Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<https://www.apache.org/>).

1.136 protocol-buffers 3.11.4

1.136.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// Copyright 2008 Google Inc. All rights reserved.  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions are  
// * Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// * Redistributions in binary form must reproduce the above  
// copyright notice, this list of conditions and the following disclaimer  
// in the documentation and/or other materials provided with the  
// * Neither the name of Google Inc. nor the names of its  
// this software without specific prior written permission.
```

Found in path(s):

```
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-  
jar/google/protobuf/source_context.proto  
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-jar/google/protobuf/field_mask.proto  
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-jar/google/protobuf/empty.proto  
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-jar/google/protobuf/timestamp.proto  
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-jar/google/protobuf/struct.proto  
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-jar/google/protobuf/wrappers.proto  
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-jar/google/protobuf/api.proto  
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-jar/google/protobuf/any.proto  
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-jar/google/protobuf/duration.proto  
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-  
jar/google/protobuf/compiler/plugin.proto  
* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-jar/google/protobuf/descriptor.proto
```

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Automatic-Module-Name: com.google.protobuf

Bnd-LastModified: 1581715154656

Build-Jdk: 1.8.0_181-google-v7

Built-By: rafikamal

Bundle-Description: Core Protocol Buffers library. Protocol Buffers are a way of encoding structured data in an efficient yet extensible format.

Bundle-DocURL: <https://developers.google.com/protocol-buffers/>

Bundle-License: <https://opensource.org/licenses/BSD-3-Clause>

Bundle-ManifestVersion: 2

Bundle-Name: Protocol Buffers [Core]

Bundle-SymbolicName: com.google.protobuf

Bundle-Version: 3.11.4

Created-By: Apache Maven Bundle Plugin
Export-Package: com.google.protobuf;version="3.11.4"
Import-Package: sun.misc;resolution:=optional,com.google.protobuf;version="[3.11,4)"
Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.7))"
Tool: Bnd-3.0.0.201509101326

Found in path(s):

* /opt/cola/permits/1063969100_1666051531.939509/0/protobuf-java-3-11-4-jar/META-INF/MANIFEST.MF

1.137 byte-buddy-agent 1.11.3

1.137.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright $\{\text{project.inceptionYear}\}$ - $\{\text{current.year}\}$ $\{\text{copyright.holder}\}$

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.138 guava v30.0

1.138.1 Available under license :

Doug Lea

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.139 aws-sdk-for-java-core 1.12.29

1.139.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the

Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.140 aws-java-sdk-regions 2.17.52

1.140.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed
with the accompanying software if such terms are included in the directory containing the accompanying software.
Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by
Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.141 junit-platform-junit-platform-commons

1.8.2

1.141.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and

* **b)** in the case of each subsequent Contributor:

* **i)** changes to the Program, and

* **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

****b)**** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

****i)**** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

****ii)**** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

****iii)**** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

****iv)**** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

****3.2**** When the Program is Distributed as Source Code:

* ****a)**** it must be made available under this Agreement, or if the Program ****i)**** is combined with other material in a separate file or files made available under a Secondary License, and ****ii)**** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* ****b)**** a copy of this Agreement must be included with each copy of the Program.

****3.3**** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: ****a)**** promptly notify the Commercial Contributor in writing of such claim, and ****b)**** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.142 beorn7-perks v1.0.1

1.142.1 Available under license :

Copyright (C) 2013 Blake Mizerany

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.143 junit 4.13.2

1.143.1 Available under license :

JUnit

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

- a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
- b) in the case of each subsequent Contributor:
 - i) changes to the Program, and
 - ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

- a) it complies with the terms and conditions of this Agreement; and
- b) its license agreement:
 - i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 - ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 - iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

- a) it must be made available under this Agreement; and
- b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial

Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

1.144 netty/transport/classes/kqueue

4.1.72.Final

1.144.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2021 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
```

Found in path(s):

* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/KQueueDomainDatagramChannel.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/AbstractKQueueDatagramChannel.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/KQueueDomainDatagramChannelConfig.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2016 The Netty Project  
*  
* The Netty Project licenses this file to you under the Apache License,  
* version 2.0 (the "License"); you may not use this file except in compliance  
* with the License. You may obtain a copy of the License at:  
*  
* https://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT  
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the  
* License for the specific language governing permissions and limitations  
* under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/KQueueSocketChannel.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/KQueueServerDomainSocketChannel.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/NativeLongArray.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/KQueueChannelOption.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/KQueue.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/KQueueSocketChannelConfig.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/KQueueDomainSocketChannelConfig.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/AbstractKQueueServerChannel.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/AbstractKQueueStreamChannel.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/AcceptFilter.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-jar/io/netty/channel/kqueue/KQueueEventLoopGroup.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-

jar/io/netty/channel/kqueue/package-info.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueEventArray.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueServerSocketChannel.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueEventLoop.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/Native.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/AbstractKQueueChannel.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueChannelConfig.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueDatagramChannelConfig.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueServerChannelConfig.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueDatagramChannel.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/BsdSocket.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueDomainSocketChannel.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueRecvByteAllocatorHandle.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueStaticallyReferencedJniMethods.java
* /opt/cola/permits/1411866943_1662690041.3508255/0/netty-transport-classes-kqueue-4-1-72-final-sources-1-
jar/io/netty/channel/kqueue/KQueueServerSocketChannelConfig.java
No license file was found, but licenses were detected in source scan.

```
<!--  
~ Copyright 2021 The Netty Project  
~  
~ The Netty Project licenses this file to you under the Apache License,  
~ version 2.0 (the "License"); you may not use this file except in compliance  
~ with the License. You may obtain a copy of the License at:  
~  
~ https://www.apache.org/licenses/LICENSE-2.0  
~  
~ Unless required by applicable law or agreed to in writing, software  
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT  
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the  
~ License for the specific language governing permissions and limitations  
~ under the License.  
-->
```

Found in path(s):

1.145 google-gson 2.8.6

1.145.1 Available under license :

Google Gson

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2008-2011 Google Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.146 aws-java-sdk-http-client-interface

2.17.52

1.146.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software.

Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by
Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.147 go-tomb-tomb 20150422-snapshot-dd632973

1.147.1 Available under license :

tomb - support for clean goroutine termination in Go.

Copyright (c) 2010-2011 - Gustavo Niemeyer <gustavo@niemeyer.net>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.148 dgraph-io/ristretto v0.1.0

1.148.1 Available under license :

bbloom.go

// The MIT License (MIT)

// Copyright (c) 2014 Andreas Briese, eduToolbox@Bri-C GmbH, Sarstedt

// Permission is hereby granted, free of charge, to any person obtaining a copy of
// this software and associated documentation files (the "Software"), to deal in
// the Software without restriction, including without limitation the rights to
// use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
// the Software, and to permit persons to whom the Software is furnished to do so,
// subject to the following conditions:

// The above copyright notice and this permission notice shall be included in all
// copies or substantial portions of the Software.

// THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
// IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
// FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
// COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
// IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
// CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

rtutil.go

// MIT License

// Copyright (c) 2019 Ewan Chou

// Permission is hereby granted, free of charge, to any person obtaining a copy
// of this software and associated documentation files (the "Software"), to deal
// in the Software without restriction, including without limitation the rights
// to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
// copies of the Software, and to permit persons to whom the Software is
// furnished to do so, subject to the following conditions:

// The above copyright notice and this permission notice shall be included in all
// copies or substantial portions of the Software.

// THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
// IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
// FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
// AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
// LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
// OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
// SOFTWARE.

Modifications:

/*

* Copyright 2019 Dgraph Labs, Inc. and Contributors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.149 protocol-buffers 3.19.3

1.149.1 Available under license :

Copyright 2008 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.
This file contains a list of people who have made large contributions to the public version of Protocol Buffers.

Original Protocol Buffers design and implementation:

Sanjay Ghemawat <sanjay@google.com>
Jeff Dean <jeff@google.com>
Daniel Dulitz <daniel@google.com>
Craig Silverstein
Paul Haahr <haahr@google.com>
Corey Anderson <corin@google.com>
(and many others)

Proto2 C++ and Java primary author:

Kenton Varda <kenton@google.com>

Proto2 Python primary authors:

Will Robinson <robinson@google.com>
Petar Petrov <petar@google.com>

Java Nano primary authors:

Brian Duff <bduff@google.com>
Tom Chao <chaot@google.com>
Max Cai <maxtroy@google.com>
Ulas Kirazci <ulas@google.com>

Large code contributions:

Jason Hsueh <jasonh@google.com>
Joseph Schorr <jschorr@google.com>
Wenbo Zhu <wenboz@google.com>

Large quantity of code reviews:

Scott Bruce <sbruce@google.com>
Frank Yellin
Neal Norwitz <nnorwitz@google.com>
Jeffrey Yasskin <jyasskin@google.com>
Ambrose Feinstein <ambrose@google.com>

Documentation:

Lisa Carey <lcarey@google.com>

Maven packaging:

Gregory Kick <gak@google.com>

Patch contributors:

Kevin Ko <kevin.s.ko@gmail.com>

* Small patch to handle trailing slashes in --proto_path flag.

Johan Euphrosine <proppy@aminche.com>

* Small patch to fix Python CallMethod().

Ulrich Kunitz <kune@deine-taler.de>

* Small optimizations to Python serialization.

Leandro Lucarella <llucax@gmail.com>

* VI syntax highlighting tweaks.

* Fix compiler to not make output executable.

Dilip Joseph <dilip.antony.joseph@gmail.com>

* Heuristic detection of sub-messages when printing unknown fields in text format.

Brian Atkinson <nairb774@gmail.com>

* Added @Override annotation to generated Java code where appropriate.

Vincent Choinire <Choiniere.Vincent@hydro.qc.ca>

* Tru64 support.

Monty Taylor <monty.taylor@gmail.com>

* Solaris 10 + Sun Studio fixes.

Alek Storm <alek.storm@gmail.com>

* Slicing support for repeated scalar fields for the Python API.

Oleg Smolsky <oleg.smolsky@gmail.com>

* MS Visual Studio error format option.

* Detect unordered_map in stl_hash.m4.

Brian Olson <brianolson@google.com>

* gzip/zlib I/O support.

Michael Poole <mdpoole@troilus.org>

* Fixed warnings about generated constructors not explicitly initializing all fields (only present with certain compiler settings).

* Added generation of field number constants.

Wink Saville <wink@google.com>

* Fixed initialization ordering problem in logging code.

Will Pierce <willp@nuclei.com>

* Small patch improving performance of in Python serialization.

Alexandre Vassalotti <alexandre@peadrop.com>

* Emacs mode for Protocol Buffers (editors/protobuf-mode.el).

Scott Stafford <scott.stafford@gmail.com>

* Added Swap(), SwapElements(), and RemoveLast() to Reflection interface.

Alexander Melnikov <alm@sibmail.ru>

* HPUX support.

Oliver Jowett <oliver.jowett@gmail.com>

* Detect whether zlib is new enough in configure script.

* Fixes for Solaris 10 32/64-bit confusion.

Evan Jones <evanj@mit.edu>

* Optimize Java serialization code when writing a small message to a stream.

- * Optimize Java serialization of strings so that UTF-8 encoding happens only once per string per serialization call.
- * Clean up some Java warnings.
- * Fix bug with permanent callbacks that delete themselves when run.

Michael Kucharski <m.kucharski@gmail.com>

- * Added CodedInputStream.getTotalBytesRead().

Kacper Kowalik <xarthisius.kk@gmail.com>

- * Fixed m4/acx_pthread.m4 problem for some Linux distributions.

William Orr <will@worrbase.com>

- * Fixed detection of sched_yield on Solaris.
- * Added atomicops for Solaris

Andrew Paprocki <andrew@ishiboo.com>

- * Fixed minor IBM xLC compiler build issues
- * Added atomicops for AIX (POWER)

1.150 gomemcache 20190913-snapshot-a41fca85

1.150.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,

including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf

of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.151 apache-commons-digester 1.8.1

1.151.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Digester

Copyright 2001-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

1.152 aws-sdk-for-go v1.40.54

1.152.1 Available under license :

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

AWS SDK for Go

Copyright 2015 Amazon.com, Inc. or its affiliates. All Rights Reserved.

Copyright 2014-2015 Stripe, Inc.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.153 assertj-fluent-assertions 3.22.0

1.153.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
 * with
 * the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the
 * specific language governing permissions and limitations under the License.
 *
 * Copyright 2012-2021 the original author or authors.
 */
/**
 * Base class for all implementations of assertions for { @link File}s.
 *
 * @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
 *     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
 * implementation</a>&quot;
 *     for more details.
 *
 * @author David DIDIER
 * @author Yvonne Wang
 * @author Alex Ruiz
 * @author Olivier Michallat
 * @author Olivier Demeijer
 * @author Mikhail Mazursky
 * @author Jean-Christophe Gay
 * @author Valeriy Vyrva
 * @author Nikolaos Georgiou
 */
```

Found in path(s):

```
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractFileAssert.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
 * with
```

```
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for { @link Float}s.
*
* @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
*     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation</a>&quot;
*     for more details.
*
* @author Drummond Dawson
* @author Yvonne Wang
* @author Alex Ruiz
* @author Ansgar Konermann
* @author Mikhail Mazursky
* @author Nicolas François
* @author Jin Kwon
*/
```

Found in path(s):

```
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractFloatAssert.java
```

No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
```

```
* Base class for all implementations of assertions for { @code CharSequence}s.
*
* @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
*     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation</a>&quot;
*     for more details.
* @param <ACTUAL> the type of the "actual" value.
*
* @author Yvonne Wang
* @author David DIDIER
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author Nicolas Francois
* @author Daniel Weber
*/
```

Found in path(s):

```
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractCharSequenceAssert.java
```

No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
```

```
/**
* Base class for all implementations of assertions for { @link Short}s.
*
* @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
*     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation</a>&quot;
*     for more details.
*
* @author Drummond Dawson
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
*/
```

* @author Alex Ruiz
* @author Mikhail Mazursky
* @author Nicolas François
* @author Cal027
*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractShortAssert.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with

* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the

* specific language governing permissions and limitations under the License.

*

* Copyright 2012-2021 the original author or authors.

*/

/**

* Base class for all implementations of assertions for { @link Double}s.

*

* @param <SELF> the "self" type of this assertion class. Please read ["http://bit.ly/1IZIRcY"](http://bit.ly/1IZIRcY)

* `target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation`

* `for more details.`

*

* @author Drummond Dawson

* @author Yvonne Wang

* @author David DIDIER

* @author Alex Ruiz

* @author Ansgar Konermann

* @author Joel Costigliola

* @author Mikhail Mazursky

* @author Nicolas François

* @author Jack Gough

*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractDoubleAssert.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
 with
 * the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 See the License for the
 * specific language governing permissions and limitations under the License.
 *
 * Copyright 2012-2021 the original author or authors.
 */
/**
 * Base class for all implementations of assertions for { @link List}s.
 * @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
 *     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
 implementation</a>&quot;
 *     for more details.
 * @param <ACTUAL> the type of the "actual" value.
 * @param <ELEMENT> the type of elements of the "actual" value.
 * @param <ELEMENT_ASSERT> used for navigational assertions to return the right assert type.
 *
 * @author Yvonne Wang
 * @author Alex Ruiz
 * @author Joel Costigliola
 * @author Mikhail Mazursky
 * @author Jacek Jackowiak
 */
```

Found in path(s):

```
*/opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractListAssert.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
 with
 * the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 See the License for the
```

```

* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for { @link Byte}s.
*
* @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
*     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation</a>&quot;
*     for more details.
*
* @author Drummond Dawson
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
* @author Alex Ruiz
* @author Mikhail Mazursky
* @author Nicolas François
* @author Cal027
*/

```

Found in path(s):

```

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractByteAssert.java

```

No license file was found, but licenses were detected in source scan.

```

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for { @link InputStream}s.
* @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
*     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation</a>&quot;
*     for more details.
* @param <ACTUAL> the type of the "actual" value.

```

*
* @author Matthieu Baechler
* @author Mikhail Mazursky
* @author Stefan Birkner
*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractInputStreamAssert.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with

* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on

* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the

* specific language governing permissions and limitations under the License.

*

* Copyright 2012-2021 the original author or authors.

*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/FutureAssert.java

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveAllNullFields.java

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/RecursiveComparisonConfiguration.java

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/AssertionErrorCreator.java

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Java6Assertions.java

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainKey.java

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/OffsetDateTimeByInstantComparator.java

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ClassAssert.java

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotMatch.java

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractCompletableFutureAssert.java

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ErrorMessageFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/AbstractComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSameClass.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractSpliteratorAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Files.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/recursive/comparison/FieldComparators.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotContainSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveBinaryContent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AssertionErrorCollector.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeUpperCase.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ElementsShouldBeExactly.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotHaveSameHashCode.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/description/Description.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Int2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeOdd.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeMarked.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveNoParent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/OptionalDoubleAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/introspection/FieldUtils.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHaveNoHost.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractIntArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeAtSameInstant.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/CommonValidations.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainExactlyInAnyOrder.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/AtomicIntegerArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/OptionalLongAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/condition/Join.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveParent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractIteratorAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/extractor/Extractors.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/TextFileWriter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainPattern.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/description/TextDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AtomicLongArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldOnlyHaveFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/recursive/comparison/DualValueDeque.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/NioFilesWrapper.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/UnsatisfiedRequirement.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldExist.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Boolean2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/IterableElementComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/JUnitBDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Urls.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AssertionInfo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Fail.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/WholeNumbers.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeRelativePath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHaveQuery.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/BDDAssumptions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/introspection/Introspection.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHavePath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Maps.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/junit/jupiter/SoftAssertionsExtension.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeEqualComparingFieldByFieldRecursively.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ObjectAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractInstantAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/IterableSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Byte2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeToday.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ElementsShouldSatisfy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/CharacterAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Iterables.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ElementsShouldHaveExactly.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeBlank.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/EnumerableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeInfinite.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/DiffAlgorithm.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainCharSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractUrlAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/ComparatorBasedComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/MapSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHavePropertyOrFieldWithValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHavePeriod.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveName.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractPredicateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualWithTimePrecision.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/BigDecimalAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Float2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveDimensions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveOnlyElementsOfType.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/introspection/MethodSupport.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/introspection/PropertySupport.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/BDDAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractDurationAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveNoSuperclass.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/BDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/recursive/comparison/ComparisonKeyDifference.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/InputStreamAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldBeCancelled.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInSameHour.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInfinite.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeUnmodifiable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/UriAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainOnlyDigits.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/AbstractOptionalAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Assertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/OptionalDoubleShouldHaveValueCloseToPercentage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeAnArray.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotHaveDuplicates.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractEnumerableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/IterableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainSequenceOfCharSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/FloatAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualIgnoringNewLineDifferences.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ArraySortedAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Short2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualIgnoringNewLines.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeFinite.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ZippedElementsShouldSatisfy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Objects.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/BooleanArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ElementsShouldHaveAtLeast.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeReadable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainExactly.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/matcher/AssertionMatcher.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/AbstractShouldHaveTextContent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeNormalized.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeRegularFile.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/configuration/Configuration.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeCanonicalPath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/recursive/comparison/RecursiveComparisonDifferenceCalculator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/ChronoLocalDateTimeComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/ClassNameComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInterface.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/CharArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeBeforeYear.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSameHashCode.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/AssertionErrorMessagesAggregator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Closeables.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/condition/DoesNotHave.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/DefaultAssertionErrorCollector.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveRootCause.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeInstanceOfAny.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Float2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotHave.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/NotThrownAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Booleans.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AssertionsForInterfaceTypes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/condition/AllOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/DurationAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Boolean2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualNormalizingUnicode.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/LongAdderAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSameHourAs.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/array2d/Array2dElementShouldBeDeepEqual.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeBefore.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractBooleanArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/BasicErrorMessageFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/IntArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeAbstract.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/WithThrowable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveMessageFindingMatchRegex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInSameHourWindow.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveNext.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/SoftProxies.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualToIgnoringFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveCauseReference.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveRootCauseInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/LocalDateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldBeCompletedExceptionally.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/WithAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInTheFuture.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldHaveFailedWithin.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/filter/InFilter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotContainCharSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Doubles.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldBeSame.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEven.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotHaveToString.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainKeys.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Array2DAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/condition/MappedCondition.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/presentation/UnicodeRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContain.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSuppressedException.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/presentation/StandardRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/BigIntegerAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Arrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInSameMonth.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/LocalTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ClassBasedNavigableIterableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/description/JoinDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotEndWith.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/DateUtil.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeCloseTo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/data/Offset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/GenericComparableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/SoftAssertionsStatement.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveRootCauseExactlyInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Long2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/internal/DigestDiff.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveExtension.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Uris.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/iterable/ThrowingExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainAnyOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Double2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveAnnotations.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeSubstring.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ListAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/BinaryDiffResult.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/myers/Snake.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveCauseExactlyInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/NoElementsShouldSatisfy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldNotContainAnyWhitespaces.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ShortArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualWithinPercentage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/uri/ShouldHaveAnchor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/data/MapEntry.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/ErrorCollector.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/InstanceOfAssertFactories.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/IterableUtil.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/DeleteDelta.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/StandardSoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/util/Strings.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractFloatArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/exception/PathsException.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ObjectArrayElementComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSameSizeAs.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/ArrayWrapperList.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AssertDelegateTarget.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/OptionalShouldBePresent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSameDimensionsAs.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractStringAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainOnlyKeys.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/diff/DiffUtils.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractTemporalAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/filter/Filters.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/DoublePredicateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/CharArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeSymbolicLink.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ElementsSatisfyingConsumer.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Digests.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractLocalDateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/FieldLocation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveScale.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeInSameSecondWindow.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/ErrorMessage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldContainSubsequenceOfCharSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/NoElementsShouldMatch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualIgnoringHours.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveAtLeastOneElementOfType.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/StandardComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveCause.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainOneOrMoreWhitespaces.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/myers/MyersDiff.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/myers/PathNode.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/FloatComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeBeforeOrEqualTo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractOffsetTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Java6JUnitBDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/junit/jupiter/InjectSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Throwables.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractAtomicFieldUpdaterAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeGreater.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBePeriod.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractDoubleArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Dates.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldSatisfy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractPeriodAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInSameMinute.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/introspection/FieldSupport.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/internal/DoubleArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSameTime.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/filter/NotInFilter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHavePropertyOrField.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldNotBeCompletedExceptionally.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/AtomicReferenceArrayElementComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldAccept.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/groups/FieldsOrPropertiesExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/OffsetDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Java6SoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveTime.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractByteArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/SoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveNoExtension.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/BooleanArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AtomicReferenceAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveNoSuppressedExceptions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Characters.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotStartWith.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/DescribableComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/OptionalShouldContainInstanceOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractBigIntegerAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ByteArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldBeSubsetOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Char2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Files.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/DeltaComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/CanIgnoreReturnValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/IndexedObjectEnumerableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Lists.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Arrays2D.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualWithinOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractPredicateLikeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSize.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/condition/AnyOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/HamcrestCondition.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHaveHost.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/TypeComparators.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AtomicLongFieldUpdaterAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeAfterOrEqualTo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeIn.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldBeEqualToWithSortedQueryParameters.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/InputStreams.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/LocalDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/FailureMessages.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/description/LazyTextDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInSameMinuteWindow.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldEndWith.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/OptionalShouldContain.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHavePort.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInstanceOfAny.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/OptionalShouldBeEmpty.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/OptionalIntAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/AnyElementShouldMatch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ClassBasedNavigableListAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AssertionsForClassTypes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/CheckReturnValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ZonedDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Java6BDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AtomicReferenceArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AtomicBooleanAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/SoftAssertionsRule.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/data/TemporalUnitLessThanOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/description/EmptyTextDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Assumptions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/LongArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldEndWithPath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractIterableSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSameContent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Abstract2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ConstructorInvoker.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/WritableAssertionInfo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/presentation/Representation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldNotHave.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEmpty.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveToString.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveDigest.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractOffsetDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/data/TemporalUnitWithinOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAssignableFrom.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ElementsShouldBe.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldContainOnlyNulls.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/Short2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/JUnitSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldHaveSizeLessThanOrEqualTo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Comparables.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeEqualNormalizingPunctuationAndWhitespace.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Lists.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/FactoryBasedNavigableListAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldBeAfter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/internal/Byte2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ShouldOnlyHaveElementsOfTypes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/error/ClassModifierShouldBe.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/util/AbstractComparableNumberComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/recursive/comparison/ComparisonDifference.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/OptionalAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/ChronoZonedDateTimeByInstantComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ByteAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractLocalDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotContainAtIndex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/SubarraysShouldHaveSameSize.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/SpliteratorAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractCharArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/InsertDelta.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AtomicIntegerAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractZonedDateTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/recursive/comparison/FieldMessages.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeLess.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/UnambiguousRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AssertProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotContain.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainSubsequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/VisibleForTesting.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveDuration.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractObjectArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Predicates.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldBeDone.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/BooleanAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Throwables.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldBeInThePast.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeLowerCase.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotExist.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AutoCloseableSoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Failures.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/condition/Negative.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/TypeHolder.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualIgnoringSeconds.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/TypeMessages.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeAtIndex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/ChangeDelta.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/MultipleAssertionsError.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/BinaryDiff.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldBeCompletedWithin.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractLongArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHaveParameter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualIgnoringTimezone.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/IteratorAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ThrowableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/IntegerAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Floats.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldBeCompleted.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractFileSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldNotBe.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/junit/jupiter/SoftlyExtension.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/BigIntegers.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractLocalTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/CollectionAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractMapSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/IterableDiff.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/presentation/HexadecimalRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/introspection/ClassUtils.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/MapAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeEmpty.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ShortAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/Chunk.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/annotations/Beta.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ClassLoadingStrategyFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHavePackage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/Warning.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Sets.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeInSameSecond.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveMethods.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/SoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/DoubleArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Long2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/ThrowableAssertAlternative.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ThrowableTypeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldMatch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractAtomicReferenceAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ComparableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/AssertJMultipleFailuresError.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeFinite.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeMixedCase.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ComparatorFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AssertJProxySetup.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractUriAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/condition/Not.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualIgnoringMinutes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Java6JUnitSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldHaveFailed.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/PathAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSuperclass.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHave.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AfterAssertionErrorCollected.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/GroupTypeDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveReference.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/myers/DiffNode.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Maps.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldBeWritable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AutoCloseableBDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/MessageFormatter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualIgnoringWhitespace.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveMessageMatchingRegex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEmptyDirectory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotContainValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/AssertionErrorMessagesAggregator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/IntPredicateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/data/TemporalUnitOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AtomicReferenceFieldUpdaterAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeAfterYear.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Descriptable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ThrowingConsumer.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldStartWith.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/extractor/ByNameSingleExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/presentation/BinaryRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveStamp.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Java6BDDSoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractComparableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSizeGreaterThanOrEqualTo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveExactlyTypes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/Delta.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/CompletableFutureAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/Int2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/data/Index.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/ByteArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/presentation/CompositeRepresentation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/ExtendedByTypesComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Char2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AutoCloseableSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeEqual.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/ShortArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/configuration/Services.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainEntries.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeEqualWithinPercentage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Java6BDDAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/DeepDifference.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotContainPattern.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/PeriodAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ElementsShouldMatch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeExactlyInstanceOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeExhausted.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/PredicateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Double2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractShortArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/FloatArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeNull.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldNotMatchPattern.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeTrue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldNotHaveFailed.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Preconditions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/OffsetTimeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/iterable/Extractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainNull.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/IntArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AssertFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AssumptionExceptionFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotAccept.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractOptionalLongAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldMatchPattern.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveNoFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/BigIntegerComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Arrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/InstanceOfAssertFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/FloatingPointNumberAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/presentation/NumberGrouping.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Condition.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeEqualNormalizingWhitespace.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeFile.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ConditionAndGroupGenericParameterTypeShouldBeTheSame.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeEqualIgnoringWhitespace.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldBeInSameYear.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/myers/Equalizer.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/groups/Tuple.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeBlank.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeSorted.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AtomicLongAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/FieldByFieldComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/RealNumbers.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveLineCount.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeAbsolutePath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/NaturalOrderComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualByComparingFieldByFieldRecursively.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBe.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/ConfigurableRecursiveFieldByFieldComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/DoubleComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Paths.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Longs.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/filter/NotFilter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainCharSequenceOnlyOnce.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/FileSizeAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/SoftAssertionError.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractOptionalDoubleAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/recursive/comparison/FieldHolder.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeOfClassIn.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldBeBase64.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/condition/VerboseCondition.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Object2DArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSizeLessThan.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ElementsShouldBeAtMost.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainSequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/BDDSoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ObjectArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/BigDecimals.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainValues.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/TriFunction.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldNotBeCancelled.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Diff.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Shorts.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ExtensionPoints.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/data/TemporalOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeExactlyInstanceOf.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/OptionalDoubleShouldHaveValueCloseToOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/OnFieldsComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeBetween.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHaveFragment.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/FactoryBasedNavigableIterableAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/DateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/introspection/MemberUtils.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldHaveSizeGreaterThan.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Streams.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/NullSafeComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/LongArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Conditions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/StringAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeAnnotation.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/LongAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/DoubleAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveContent.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeIn.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Spliterators.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualByComparingOnlyGivenFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/PathNaturalOrderComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Hexadecimals.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Futures.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainRecursively.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotContainKeys.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeGreaterOrEqual.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldNotBeCompleted.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/InputStreamsException.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/introspection/IntrospectionError.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Java6StandardSoftAssertionsProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/recursive/comparison/DualValue.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/AtomicIntegerFieldUpdaterAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/CommonErrors.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/JUnitJupiterBDDSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ElementsShouldBeAtLeast.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainEntry.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/FileAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AtomicStampedReferenceAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeLessOrEqual.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeNullOrEmpty.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveCauseInstance.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Assert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotHaveThrown.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqual.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/groups/Properties.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/UrlAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainsOnlyOnce.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AtomicMarkableReferenceAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Classes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/InstantAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualIgnoringNanos.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/xml/XmlStringPrettyFormatter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/configuration/PreferredAssumptionException.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotContainNull.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/data/Percentage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/extractor/ResultOfExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Iterators.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/WithAssumptions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/AssertionErrorFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/LongPredicateAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ElementsShouldNotBe.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotContainSubsequence.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualNormalizingWhitespace.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainOnlyWhitespaces.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/JUnitJupiterSoftAssertions.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotContainOnlyWhitespaces.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/BigDecimalComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/configuration/ConfigurationProvider.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractFutureAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/CharSequenceAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveMessage.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ProxifyMethodChangingTheObjectUnderTest.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHaveScheme.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Bytes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractPathAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/Objects.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ElementsShouldHave.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/Object2DArrayAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeDirectory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/error/ShouldBeInSameDay.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Numbers.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveSizeBetween.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHaveAuthority.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/introspection/PropertyOrFieldSupport.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/ObjectArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainAtIndex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveDateField.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeFalse.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeExecutable.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeSame.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/future/ShouldNotBeDone.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeBetween.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/NumberAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveNoCause.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotHaveAnyElementsOfTypes.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/diff/Patch.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractOptionalIntAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeEqualIgnoringCase.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Strings.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/filter/FilterOperator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainKey.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHaveUserInfo.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/extractor/ToStringExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/internal/IgnoringFieldsComparator.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeEqualWithinOffset.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveAtIndex.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/presentation/PredicateDescription.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/extractor/ByNameMultipleExtractor.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotHaveSameClass.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldContainOnly.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldNotBeOfClassIn.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/DescriptionFormatter.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/Integers.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldBeEqualIgnoringCase.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/FloatArrays.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ObjectAssertFactory.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/internal/ComparisonStrategy.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldHaveNonNullFields.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/util/URLs.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/uri/ShouldHaveProtocol.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ShouldStartWithPath.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/RecursiveComparisonAssert.java
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/error/ElementsShouldHaveAtMost.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.

*
* Copyright 2012-2021 the original author or authors.

*/
/**

* Base class for all assertions.

*

* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
* target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation"

* for more details.

* @param <ACTUAL> the type of the "actual" value.

*

* @author Alex Ruiz

* @author Joel Costigliola

* @author Mikhail Mazursky

* @author Nicolas François

*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractAssert.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with

* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the

* specific language governing permissions and limitations under the License.

*

* Copyright 2012-2021 the original author or authors.

*/

/**

* Base class for all implementations of assertions for { @link Boolean}s.

*

* @param <SELF> the "self" type of this assertion class. Please read "<a href="http://bit.ly/1IZIRcY"
* target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation"

* for more details.

*
* @author Alex Ruiz
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
* @author Mikhail Mazursky
*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractBooleanAssert.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with

* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the

* specific language governing permissions and limitations under the License.

*

* Copyright 2012-2021 the original author or authors.

*/

/**

* Base class for all implementations of assertions for { @link Object}s.

*

* @param <SELF> the "self" type of this assertion class. Please read ["http://bit.ly/1IZIRcY"](http://bit.ly/1IZIRcY) [target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation](#)

* for more details.

* @param <ACTUAL> the type of the "actual" value.

*

* @author Yvonne Wang

* @author Alex Ruiz

* @author Nicolas François

* @author Mikhail Mazursky

* @author Joel Costigliola

* @author Libor Ondrusek

*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractObjectAssert.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
 with
 * the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 See the License for the
 * specific language governing permissions and limitations under the License.
 *
 * Copyright 2012-2021 the original author or authors.
 */
/**
 * Assertions methods applicable to groups of objects (e.g. arrays or collections.)
 *
 * @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
 * target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
 implementation</a>&quot;
 * for more details.
 * @param <ELEMENT> the type of elements of the "actual" value.
 *
 * @author Yvonne Wang
 * @author Alex Ruiz
 * @author Nicolas François
 * @author Mikhail Mazursky
 * @author Joel Costigliola
 * @author Nicolas François
 * @author Florent Biville
 */
```

Found in path(s):

```
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/ObjectEnumerableAssert.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
 with
 * the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

See the License for the

* specific language governing permissions and limitations under the License.

*

* Copyright 2012-2021 the original author or authors.

*/

/**

* Base class for all implementations of assertions for { @link Date}s.

* <p>

* Note that assertions with date parameter comes with two flavor, one is obviously a { @link Date} and the other is a

* { @link String} representing a Date.

* For the latter, the default format follows ISO 8901 : "yyyy-MM-dd", user can override it with a custom format by

* calling { @link #withDateFormat(DateFormat)}.

* The user custom format will then be used for all next Date assertions (i.e not limited to the current assertion) in

* the test suite.

* To turn back to default format, simply call { @link #withDefaultDateFormatsOnly()}.

*

* @param <SELF> the "self" type of this assertion class. Please read "

* target='_blank'>Emulating 'self types' using Java Generics to simplify fluent API implementation" for

* more details.

* @author Tomasz Nurkiewicz (thanks for giving assertions idea)

* @author Joel Costigliola

* @author Mikhail Mazursky

* @author William Delanoue

* @author Michal Kordas

* @author Eddú Meléndez

*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-

jar/org/assertj/core/api/AbstractDateAssert.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with

* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on

* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the

* specific language governing permissions and limitations under the License.

*

* Copyright 2012-2021 the original author or authors.

*/

/**

* Base class for all implementations of assertions for { @link Integer}s.

*
* @param <SELF> the "self" type of this assertion class. Please read ["target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation](http://bit.ly/1IZIRcY)
* for more details.
*
* @author Drummond Dawson
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author Nicolas François
* @author Cal027
*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractIntegerAssert.java

No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* <http://www.apache.org/licenses/LICENSE-2.0>
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/

/**
* Base class for all implementations of assertions for { @link Map}s.
*
* @param <SELF> the "self" type of this assertion class. Please read ["target="_blank">Emulating 'self types' using Java Generics to simplify fluent API implementation](http://bit.ly/1IZIRcY)
* for more details.
* @param <ACTUAL> the type of the "actual" value.
* @param <K> the type of keys in the map.
* @param <V> the type of values in the map.
*
* @author David DIDIER

* @author Yvonne Wang
* @author Alex Ruiz
* @author Mikhail Mazursky
* @author Nicolas François
* @author dorzey
* @author Filip Hrisafov
*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractMapAssert.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with

* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the

* specific language governing permissions and limitations under the License.

*

* Copyright 2012-2021 the original author or authors.

*/

/**

* Base class for all implementations of assertions for { @link Collection}s.

* @param <SELF> the "self" type of this assertion class. Please read ["http://bit.ly/1IZIRcY"](http://bit.ly/1IZIRcY)

* [target="_blank">Emulating 'self types' using Java Generics to simplify fluent API](#)

implementation"

* [for more details.](#)

* @param <ACTUAL> the type of the "actual" value.

* @param <ELEMENT> the type of elements of the "actual" value.

* @param <ELEMENT_ASSERT> used for navigational assertions to return the right assert type.

*

* @since 3.21.0

*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractCollectionAssert.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with

* the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for { @link Throwable}s.
*
* @param <SELF> the "self" type of this assertion class. Please read "Emulating 'self types' using Java Generics to simplify fluent API
* implementation"
* for more details.
* @param <ACTUAL> the type of the "actual" value.
*
* @author David DIDIER
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
* @author Jack Gough
* @author Mike Gilchrist
*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractThrowableAssert.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with

* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the

* specific language governing permissions and limitations under the License.

*

* Copyright 2012-2021 the original author or authors.

*/

/**

```
* Base class for all implementations of assertions for { @link Class }es.
*
* @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
*     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation</a>&quot;;
*     for more details.
*
* @author William Delanoue
* @author Mikhail Mazursky
*/
```

Found in path(s):

```
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractClassAssert.java
```

No license file was found, but licenses were detected in source scan.

```
/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
```

```
* Base class for implementations of <code>{@link ObjectEnumerableAssert}</code> whose actual value type is
* <code>{@link Collection}</code>.
```

```
*
* @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
*     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
implementation</a>&quot;;
*     for more details.
* @param <ACTUAL> the type of the "actual" value.
* @param <ELEMENT> the type of elements of the "actual" value.
* @param <ELEMENT_ASSERT> used for navigational assertions to return the right assert type.
*
```

```
* @author Yvonne Wang
* @author Alex Ruiz
* @author Mathieu Baechler
* @author Joel Costigliola
* @author Maciej Jaskowski
* @author Nicolas François
```

* @author Mikhail Mazursky
* @author Mateusz Haligowski
* @author Lovro Pandzic
* @author Marko Bekhta
*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractIterableAssert.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with

* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on

* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the

* specific language governing permissions and limitations under the License.

*

* Copyright 2012-2021 the original author or authors.

*/

/**

* Base class for all implementations of assertions for { @link BigDecimal}s.

*

* @param <SELF> the "self" type of this assertion class. Please read ["http://bit.ly/1IZIRcY"](http://bit.ly/1IZIRcY)

* `target="_blank">Emulating 'self types' using Java Generics to simplify fluent API`

implementation"

* `for more details.`

*

* @author Drummond Dawson

* @author David DIDIER

* @author Ted M. Young

* @author Yvonne Wang

* @author Alex Ruiz

* @author Joel Costigliola

* @author Mikhail Mazursky

*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractBigDecimalAssert.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/
/**
* Base class for all implementations of assertions for { @link Character}s.
*
* @param <SELF> the "self" type of this assertion class. Please read ["http://bit.ly/1IZIRcY"](http://bit.ly/1IZIRcY)
* `target="_blank">Emulating 'self types' using Java Generics to simplify fluent API`
implementation"
* `for more details.`
*
* @author Yvonne Wang
* @author David DIDIER
* @author Ansgar Konermann
* @author Alex Ruiz
* @author Joel Costigliola
* @author Mikhail Mazursky
*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractCharacterAssert.java

No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the
* specific language governing permissions and limitations under the License.
*
* Copyright 2012-2021 the original author or authors.
*/

```
/**
 * Base class for all implementations of assertions for { @link Long}s.
 *
 * @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
 *     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
 * implementation</a>&quot;
 *     for more details.
 *
 *
 * @author Drummond Dawson
 * @author Yvonne Wang
 * @author David DIDIER
 * @author Ansgar Konermann
 * @author Alex Ruiz
 * @author Joel Costigliola
 * @author Mikhail Mazursky
 * @author Nicolas François
 * @author Cal027
 */
```

Found in path(s):

```
* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-
jar/org/assertj/core/api/AbstractLongAssert.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
 * with
 * the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the
 * specific language governing permissions and limitations under the License.
 *
 * Copyright 2012-2021 the original author or authors.
 */
```

```
/**
 * Base class for all implementations of assertions for { @link LongAdder}s.
 *
 * @param <SELF> the "self" type of this assertion class. Please read &quot;<a href="http://bit.ly/1IZIRcY"
 *     target="_blank">Emulating 'self types' using Java Generics to simplify fluent API
 * implementation</a>&quot;
 *     for more details.
 *
 *
 * @author Grzegorz Piwowarek
 * @since 3.16.0
 */
```

*/

Found in path(s):

* /opt/cola/permits/1341130270_1654817020.9922328/0/assertj-core-3-22-0-sources-jar/org/assertj/core/api/AbstractLongAdderAssert.java

1.154 cespare-xxhash v1.1.0

1.154.1 Available under license :

Copyright (c) 2016 Caleb Spare

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.155 x-xerrors 20200804-snapshot-5ec99f83

1.155.1 Available under license :

Copyright (c) 2019 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the

distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.156 apache-log4j-slf4j-binding 2.17.1

1.156.1 Available under license :

Apache Log4j SLF4J Binding
Copyright 1999-1969 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.157 golang-glog 20160125-snapshot-23def4e6

1.157.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for

the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices

that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.158 aws-java-sdk-sdk-core 2.17.52

1.158.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed
with the accompanying software if such terms are included in the directory containing the accompanying software.
Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by
Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.159 mockito-inline 3.11.2

1.159.1 Available under license :

The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.160 go-check-check 20190902-snapshot-41f04d3b

1.160.1 Available under license :

Gocheck - A rich testing framework for Go

Copyright (c) 2010-2013 Gustavo Niemeyer <gustavo@niemeyer.net>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.161 hamcrest 1.3

1.161.1 Available under license :

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
<HEAD>
<TITLE>Common Public License - v 1.0</TITLE>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</HEAD>

<BODY BGCOLOR="#FFFFFF" VLINK="#800000">

<P ALIGN="CENTER"><B>Common Public License - v 1.0</B>
<P><B></B><FONT SIZE="3"></FONT>
<P><FONT SIZE="3"></FONT><FONT SIZE="2">THE ACCOMPANYING PROGRAM IS PROVIDED
UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE,
REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE
OF THIS AGREEMENT.</FONT>
<P><FONT SIZE="2"></FONT>
<P><FONT SIZE="2"><B>1. DEFINITIONS</B></FONT>
<P><FONT SIZE="2">"Contribution" means:</FONT>

<UL><FONT SIZE="2">a) in the case of the initial Contributor, the initial code and documentation distributed
under this Agreement, and<BR CLEAR="LEFT">
b) in the case of each subsequent Contributor:</FONT></UL>

<UL><FONT SIZE="2">i) changes to the Program, and</FONT></UL>

<UL><FONT SIZE="2">ii) additions to the Program;</FONT></UL>

<UL><FONT SIZE="2">where such changes and/or additions to the Program originate from and are distributed by
that particular Contributor. </FONT><FONT SIZE="2">A Contribution 'originates' from a Contributor if it was
added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. </FONT><FONT
SIZE="2">Contributions do not include additions to the Program which: (i) are separate modules of software
distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of
```


the Program.

<P>

<P>"Contributor" means any person or entity that distributes the Program.

<P>

<P>"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

<P>

<P>"Program" means the Contributions distributed in accordance with this Agreement.

<P>

<P>"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

<P>

<P>2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

<P>3. REQUIREMENTS

<P>A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

<P>When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

<P><STRIKE></STRIKE>
<P><STRIKE></STRIKE>Contributors
may not remove or alter any copyright notices contained within the Program.

<P>

<P>Each Contributor must identify itself as the originator of its Contribution, if any, in a
manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

<P>

<P>4. COMMERCIAL DISTRIBUTION

<P>Commercial distributors of software may accept certain responsibilities with respect to end
users, business partners and the like. While this license is intended to facilitate the commercial use of the Program,
the Contributor who includes the Program in a commercial product offering should do so in a manner which does
not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a
commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify
every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses")
arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to
the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the
Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses
relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor
must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial
Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement
negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

<P>

<P>For example, a Contributor might include the Program in a commercial product offering,
Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes
performance claims, or offers warranties related to Product X, those performance claims and warranties are such
Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to
defend claims against the other Contributors related to those performance claims and warranties, and if a court
requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those
damages.

<P>

<P>5. NO
WARRANTY

<P>EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS
PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER
EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF
TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each
Recipient is solely responsible for determining the appropriateness of using and
distributing the Program and assumes all risks associated
with its exercise of rights under this Agreement, including but not limited to the risks
and costs of program errors, compliance with applicable laws, damage to or loss of data, <FONT
SIZE="2">programs or equipment, and unavailability or interruption of operations.

<P>

<P>

<P>6. DISCLAIMER OF LIABILITY

<P>EXCEPT AS EXPRESSLY SET FORTH IN THIS
AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR

ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

<P>

</BODY>

</HTML>

BSD License

Copyright (c) 2000-2006, www.hamcrest.org

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2000-2003, jMock.org

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of jMock nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
```

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>EasyMock License</title>
<link rel="stylesheet" href="easymock.css" />
</head>
```

```
<body><div class="bodywidth">
```

```
<h2>
EasyMock 2 License (MIT License)
</h2>
```

```
<em>Copyright (c) 2001-2006 <a href="http://www.offis.de">OFFIS</a>, <a
href="http://www.tammofreese.de">Tammo Freese</a>.</em>
```

```
<p>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:
```

```
</p>
<p>
```

```
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.
```

</p>

<p>

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

</p>

</div>

</body>

</html>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or

Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work

or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work

by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.162 error_prone_annotations 2.4.0

1.162.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2014 The Error Prone Authors.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-  
jar/com/google/errorprone/annotations/NoAllocation.java
```

* /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/concurrent/LockMethod.java
* /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/concurrent/UnlockMethod.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 The Error Prone Authors.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/MustBeClosed.java

* /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/FormatString.java

* /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/CompatibleWith.java

* /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/FormatMethod.java

* /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/RestrictedApi.java

* /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/DoNotMock.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2015 The Error Prone Authors.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/CompileTimeConstant.java
- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/RequiredModifiers.java
- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/CanIgnoreReturnValue.java
- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/Immutable.java
- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/SuppressPackageLocation.java
- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/ForOverride.java
- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/Var.java
- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/concurrent/LazyInit.java
- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/IncompatibleModifiers.java

No license file was found, but licenses were detected in source scan.

/*

- * Copyright 2017 The Error Prone Authors.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/DoNotCall.java
- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-jar/com/google/errorprone/annotations/OverridingMethodsMustInvokeSuper.java
- * /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-

jar/com/google/errorprone/annotations/CheckReturnValue.java
* /opt/cola/permits/1274702949_1645233644.02/0/error-prone-annotations-2-4-0-sources-
jar/com/google/errorprone/annotations/concurrent/GuardedBy.java

1.163 jetbrains-annotations 17.0.0

1.163.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2000-2016 JetBrains s.r.o.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/jetbrains/annotations/Contract.java

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2000-2014 JetBrains s.r.o.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-  
zip/org/intellij/lang/annotations/MagicConstant.java  
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-  
zip/org/jetbrains/annotations/Nullable.java
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Copyright 2000-2015 JetBrains s.r.o.
```

```
*
```

```
* Licensed under the Apache License, Version 2.0 (the "License");
```

```
* you may not use this file except in compliance with the License.
```

```
* You may obtain a copy of the License at
```

```
*
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

```
*
```

```
* Unless required by applicable law or agreed to in writing, software
```

```
* distributed under the License is distributed on an "AS IS" BASIS,
```

```
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

```
* See the License for the specific language governing permissions and
```

```
* limitations under the License.
```

```
*/
```

Found in path(s):

```
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-  
zip/org/jetbrains/annotations/TestOnly.java
```

```
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-  
zip/org/intellij/lang/annotations/Flow.java
```

```
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-  
zip/org/jetbrains/annotations/Nls.java
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Copyright 2006 Sascha Weinreuter
```

```
*
```

```
* Licensed under the Apache License, Version 2.0 (the "License");
```

```
* you may not use this file except in compliance with the License.
```

```
* You may obtain a copy of the License at
```

```
*
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

```
*
```

```
* Unless required by applicable law or agreed to in writing, software
```

```
* distributed under the License is distributed on an "AS IS" BASIS,
```

```
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

```
* See the License for the specific language governing permissions and
```

```
* limitations under the License.
```

```
*/
```

Found in path(s):

* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/intellij/lang/annotations/Identifier.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/intellij/lang/annotations/RegExp.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/intellij/lang/annotations/Subst.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/intellij/lang/annotations/Language.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/intellij/lang/annotations/Pattern.java
* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/intellij/lang/annotations/PrintFormat.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2000-2012 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/jetbrains/annotations/NotNull.java

* /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/intellij/lang/annotations/JdkConstants.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2000-2009 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/jetbrains/annotations/PropertyKey.java
- * /opt/cola/permits/1331473649_1653510392.001332/0/1463-annotations-17-0-0-sources-
zip/org/jetbrains/annotations/NonNls.java

1.164 project-lombok 1.18.12

1.164.1 Available under license :

Copyright (C) 2009-2015 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.165 apache-commons-validator 1.6

1.165.1 Available under license :

Apache Commons Validator

Copyright 2001-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.166 jackson-core 2.12.3

1.166.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or

otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents

of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.167 apiguardian-apiguardian-api 1.1.2

1.167.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"

means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and

attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the

appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.168 jackson-dataformats-binary 2.12.5

1.168.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Bnd-LastModified: 1630028340227

Build-Jdk-Spec: 1.8

Bundle-Description: Support for reading and writing Concise Binary Object Representation([CBOR](https://www.rfc-editor.org/info/rfc7049)) encoded data using Jackson abstractions (streaming API, data binding, tree model)

Bundle-DocURL: <http://github.com/FasterXML/jackson-dataformats-binary>

Bundle-License: <http://www.apache.org/licenses/LICENSE-2.0.txt>

Bundle-ManifestVersion: 2

Bundle-Name: Jackson dataformat: CBOR

Bundle-SymbolicName: com.fasterxml.jackson.dataformat.jackson-dataformat-cbor

Bundle-Vendor: FasterXML

Bundle-Version: 2.12.5

Created-By: Apache Maven Bundle Plugin

Export-Package: com.fasterxml.jackson.dataformat.cbor;uses:="com.fasterxml.jackson.core,com.fasterxml.jackson.core.base,com.fasterxml.jackson.core.format,com.fasterxml.jackson.core.io,com.fasterxml.jackson.core.json,com.fasterxml.jackson.core.sym,com.fasterxml.jackson.core.util";version="2.12.5",com.fasterxml.jackson.dataformat.cbor.databind;uses:="com.fasterxml.jackson.core,com.fasterxml.jackson.databind,com.fasterxml.jackson.databind.cfg,com.fasterxml.jackson.dataformat.cbor";version="2.12.5"

Implementation-Build-Date: 2021-08-27 01:38:53+0000

Implementation-Title: Jackson dataformat: CBOR

Implementation-Vendor: FasterXML

Implementation-Vendor-Id: com.fasterxml.jackson.dataformat

Implementation-Version: 2.12.5

Import-Package: com.fasterxml.jackson.core;version="[2.12,3)",com.fasterxml.jackson.core.base;version="[2.12,3)",com.fasterxml.jackson.core.format;version="[2.12,3)",com.fasterxml.jackson.core.io;version="[2.12,3)",com.fasterxml.jackson.core.json;version="[2.12,3)",com.fasterxml.jackson.core.sym;version="[2.12,3)",com.fasterxml.jackson.core.util;version="[2.12,3)",com.fasterxml.jackson.databind;version="[2.12,3)",com.fasterxml.jackson.databind.cfg;version="[2.12,3)",com.fasterxml.jackson.dataformat.cbor;version="[2.12,3)"

Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.7))"

Specification-Title: Jackson dataformat: CBOR

Specification-Vendor: FasterXML

Specification-Version: 2.12.5

Tool: Bnd-5.1.1.202006162103

X-Compile-Source-JDK: 1.7

X-Compile-Target-JDK: 1.7

Found in path(s):

* /opt/cola/permits/1444731879_1666051375.3560746/0/jackson-dataformat-cbor-2-12-5-jar/META-INF/MANIFEST.MF

1.169 tail v1.4.8

1.169.1 Available under license :

The MIT License (MIT)

Copyright 2015 Hewlett Packard Enterprise Development LP

Copyright (c) 2014 ActiveState

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This source code was written by the Go contributors.

The master list of contributors is in the main Go distribution,

visible at <http://tip.golang.org/CONTRIBUTORS>.

Copyright (C) 2013 99designs

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the

Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2012 The Go Authors. All rights reserved.

Copyright (c) 2012-2019 fsnotify Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

tomb - support for clean goroutine termination in Go.

Copyright (c) 2010-2011 - Gustavo Niemeyer <gustavo@niemeyer.net>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice,

this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.170 junit-jupiter-aggregator 5.8.2

1.170.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and

* **b)** in the case of each subsequent Contributor:

* **i)** changes to the Program, and

* **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

3.1 If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

* **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

* **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
* **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* **b)** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable

laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.171 io-grpc-grpc-core 1.39.0

1.171.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity

exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2016-2020 Istio Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Envoy

Copyright 2016-2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner].

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

zero-allocation-hashing

Copyright 2015 Higher Frequency Trading <http://www.higherfrequencytrading.com>

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object

form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a

file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright 2015 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Copyright 2014 The gRPC Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product contains a modified portion of 'OkHttp', an open source HTTP & SPDY client for Android and Java applications, which can be obtained at:

- * LICENSE:
 - * [okhttp/third_party/okhttp/LICENSE](#) (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/square/okhttp>
- * LOCATION_IN_GRPC:
 - * [okhttp/third_party/okhttp](#)

This product contains a modified portion of 'Envoy', an open source cloud-native high-performance edge/middle/service proxy, which can be obtained at:

- * LICENSE:
 - * [xds/third_party/envoy/LICENSE](#) (Apache License 2.0)
- * NOTICE:
 - * [xds/third_party/envoy/NOTICE](#)
- * HOMEPAGE:
 - * <https://www.envoyproxy.io>
- * LOCATION_IN_GRPC:
 - * [xds/third_party/envoy](#)

This product contains a modified portion of 'protoc-gen-validate (PGV)', an open source protoc plugin to generate polyglot message validators, which can be obtained at:

- * LICENSE:
 - * [xds/third_party/protoc-gen-validate/LICENSE](#) (Apache License 2.0)
- * NOTICE:
 - * [xds/third_party/protoc-gen-validate/NOTICE](#)
- * HOMEPAGE:
 - * <https://github.com/envoyproxy/protoc-gen-validate>
- * LOCATION_IN_GRPC:
 - * [xds/third_party/protoc-gen-validate](#)

This product contains a modified portion of 'udpa', an open source universal data plane API, which can be obtained at:

- * LICENSE:
 - * [xds/third_party/udpa/LICENSE](#) (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/cncf/udpa>
- * LOCATION_IN_GRPC:
 - * [xds/third_party/udpa](#)

Apache License

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions

to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices

stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

- 5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
- 6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
- 7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.172 go-testify 1.7.0

1.172.1 Available under license :

MIT License

Copyright (c) 2012-2020 Mat Ryer, Tyler Bunnell and contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.173 jmes-path-query-library 1.12.29

1.173.1 Available under license :

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>

Found in path(s):

* /opt/cola/permits/1411866864_1662690625.6026049/0/jmespath-java-1-12-29-jar/META-INF/maven/com.amazonaws/jmespath-java/pom.xml

1.174 docker-java-transport-zero-dep 3.2.13

1.174.1 Available under license :

Apache Commons Codec

Copyright 2002-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

`src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java`

contains test data from <http://aspell.net/test/orig/batch0.tab>.

Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

The content of package `org.apache.commons.codec.language.bm` has been translated from the original php source code available at <http://stevemorse.org/phoneticinfo.htm> with permission from the original authors.

Original source copyright:

Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

Apache HttpClient

Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.175 asm-analysis 9.1

1.175.1 Available under license :

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):

* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/package.html

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):

* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/BasicVerifier.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/Analyzer.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/SourceValue.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-


```
jar/org/objectweb/asm/tree/analysis/SimpleVerifier.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/Interpreter.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/Value.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/AnalyzerException.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/Frame.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/BasicValue.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/Subroutine.java
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/SmallSet.java
```

No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new { @link BasicInterpreter } for the latest ASM API version. <i>Subclasses must
 * not use this constructor</i>. Instead, they must use the { @link #BasicInterpreter(int) }
 * version.
 */
```

Found in path(s):

```
* /opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/BasicInterpreter.java
```

No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
```

```
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new { @link SourceInterpreter} for the latest ASM API version. <i>Subclasses must
 * not use this constructor</i>. Instead, they must use the { @link #SourceInterpreter(int)}
 * version.
 */
```

Found in path(s):

```
*/opt/cola/permits/1183891316_1627493686.82/0/asm-analysis-9-1-sources-
jar/org/objectweb/asm/tree/analysis/SourceInterpreter.java
```

1.176 testcontainers-core 1.17.3

1.176.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.177 bean-validation-api 2.0.1

1.177.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2013 Cognifide Limited

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.178 scala v2.12.10

1.178.1 Available under license :

Scala includes the JLine library:

Copyright (c) 2002-2006, Marc Prud'hommeaux <mwp1@cornell.edu>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of JLine nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
{% if site.thisScalaVersion != site.latestScalaVersion %}
```

```
<div class="version-notice">This is the specification of a previous version of Scala. See the <a href="{ { site.baseurl } }/{ { site.latestScalaVersion } }/">Scala { { site.latestScalaVersion } } spec</a>.</div>
```

```
{% endif %}
```

```
class C {
```

```
  val x: ((Int, Int) => Int) = (((a, b)) => a)
```

```
  val y: ((Int, Int, Int) => Int) = (((a, !!)) => a)
```

```
  val z: ((Int, Int, Int) => Int) = (((a, NotAPatternVariableName, c)) => a)
```

```
}
```

Scala

Copyright (c) 2002-2019 EPFL

Copyright (c) 2011-2019 Lightbend, Inc.

Scala includes software developed at
LAMP/EPFL (<https://lamp.epfl.ch/>) and
Lightbend, Inc. (<https://www.lightbend.com/>).

Licensed under the Apache License, Version 2.0 (the "License").
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This software includes projects with other licenses -- see `doc/LICENSE.md`.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and
limitations under the License.

Scala includes the ASM library.

Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

not-a-legal-formal-parameter-tuple.scala:2: error: not a legal formal parameter.

Note: Tuples cannot be directly destructured in method or function parameters.

```
    Either create a single parameter accepting the Tuple2,  
    or consider a pattern matching anonymous function: `{ case (a, b) => ... }`  
val x: ((Int, Int) => Int) = (((a, b)) => a)  
    ^
```

not-a-legal-formal-parameter-tuple.scala:3: error: not a legal formal parameter.

Note: Tuples cannot be directly destructured in method or function parameters.

```
    Either create a single parameter accepting the Tuple2,  
    or consider a pattern matching anonymous function: `{ case (param1, param2) => ... }`  
val y: ((Int, Int, Int) => Int) = (((a, !!)) => a)  
    ^
```

not-a-legal-formal-parameter-tuple.scala:4: error: not a legal formal parameter.

Note: Tuples cannot be directly destructured in method or function parameters.

```
    Either create a single parameter accepting the Tuple3,  
    or consider a pattern matching anonymous function: `{ case (param1, ..., param3) => ... }`  
val z: ((Int, Int, Int) => Int) = (((a, NotAPatternVariableName, c)) => a)  
    ^
```

three errors found

Scala is licensed under the [Apache License Version 2.0](<https://www.apache.org/licenses/LICENSE-2.0>).

Scala License

Copyright (c) 2002-2019 EPFL

Copyright (c) 2011-2019 Lightbend, Inc.

All rights reserved.

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

Other Licenses

This software includes projects with the following licenses,
which are also included in the `licenses/` directory:

[Apache License](<http://www.apache.org/licenses/LICENSE-2.0.html>)

This license is used by the following third-party libraries:

- * jansi

[BSD License](<http://www.opensource.org/licenses/bsd-license.php>)

This license is used by the following third-party libraries:

- * jline

[BSD 3-Clause License](<http://opensource.org/licenses/BSD-3-Clause>)

This license is used by the following third-party libraries:

- * asm

[MIT License](<http://www.opensource.org/licenses/MIT>)

This license is used by the following third-party libraries:

- * jquery

- * tools tooltip

package scala.build

```
import sbt._, Keys._, plugins._
```

```
object License extends AutoPlugin {
```

```
  val licenseMapping = settingKey[Seq[(File, String)]]("LICENSE/NOTICE file mappings")
```

```
  override val requires = JvmPlugin
```

```
  override val trigger = AllRequirements
```

```
  override def projectSettings: Seq[Def.Setting[_]] =
```

```
    List(packageSrc, packageBin, packageDoc)
```

```
      .map(pkg => mappings in (Compile, pkg) += licenseMapping.value)
```

```
  override def buildSettings: Seq[Def.Setting[_]] = Seq(  
    licenseMapping := List("LICENSE", "NOTICE").map(fn => (baseDirectory.value / fn) -> fn)
```

```
  )
```

```
}
```

Scala includes the Tools Tooltip library:

Copyright (c) 2009 Tero Piirainen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

(c) 2012-2014 GitHub

When using the GitHub logos, be sure to follow the GitHub logo guidelines (<https://github.com/logos>)

Font License: SIL OFL 1.1 (<http://scripts.sil.org/OFL>)

Applies to all font files

Code License: MIT (<http://choosealicense.com/licenses/mit/>)

Applies to all other files

Copyright (c) 2006, Ivan Sagalaev

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of highlight.js nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Scala includes the JLine library, which includes the Jansi library.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Scala includes the jQuery library:

Copyright (c) 2010 John Resig

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

(The MIT License)

Copyright (c) 2013 Greg Allen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.179 leaktest v1.3.0

1.179.1 Available under license :

Copyright (c) 2012 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright

notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.180 netty-transport-native-unix-common

4.1.72.Final

1.180.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2014 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License,

* version 2.0 (the "License"); you may not use this file except in compliance

* with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the

* License for the specific language governing permissions and limitations

* under the License.

*/

Found in path(s):

* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/IovArray.java

```
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/package-info.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/Unix.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/UnixChannelOption.java
No license file was found, but licenses were detected in source scan.
```

```
/*
* Copyright 2021 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/DomainDatagramChannel.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/DomainDatagramPacket.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/DomainDatagramSocketAddress.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/DomainDatagramChannelConfig.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/SegmentedDatagramPacket.java
No license file was found, but licenses were detected in source scan.
```

```
/*
* Copyright 2020 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
```

- * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
- * License for the specific language governing permissions and limitations
- * under the License.
- */

Found in path(s):

- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix.c
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix.h

No license file was found, but licenses were detected in source scan.

/*

- * Copyright 2015 The Netty Project
- *
- * The Netty Project licenses this file to you under the Apache License,
- * version 2.0 (the "License"); you may not use this file except in compliance
- * with the License. You may obtain a copy of the License at:
- *
- * <https://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
- * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
- * License for the specific language governing permissions and limitations
- * under the License.
- */

Found in path(s):

- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/NativeInetAddress.java
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/DomainSocketAddress.java
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_filedescriptor.c
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_errors.c
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_errors.h
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_filedescriptor.h
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/DomainSocketChannel.java
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_socket.h
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/DomainSocketReadMode.java
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-

1-jar/io/netty/channel/unix/Socket.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/Errors.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/UnixChannel.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_socket.c
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/DomainSocketChannelConfig.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/ServerDomainSocketChannel.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/DatagramSocketAddress.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/FileDescriptor.java
No license file was found, but licenses were detected in source scan.

```
/*  
 * Copyright 2018 The Netty Project  
 *  
 * The Netty Project licenses this file to you under the Apache License,  
 * version 2.0 (the "License"); you may not use this file except in compliance  
 * with the License. You may obtain a copy of the License at:  
 *  
 * https://www.apache.org/licenses/LICENSE-2.0  
 *  
 * Unless required by applicable law or agreed to in writing, software  
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT  
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the  
 * License for the specific language governing permissions and limitations  
 * under the License.  
 */
```

Found in path(s):

* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_buffer.h
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_buffer.c
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/PreferredDirectByteBufAllocator.java
* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/Buffer.java
No license file was found, but licenses were detected in source scan.

```
<!--  
 ~ Copyright 2016 The Netty Project  
 ~  
 ~ The Netty Project licenses this file to you under the Apache License,
```


~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ <https://www.apache.org/licenses/LICENSE-2.0>
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->

Found in path(s):

* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/META-INF/maven/io.netty/netty-transport-native-unix-common/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.

*/

Found in path(s):

* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io.netty/channel/unix/UnixChannelUtil.java

* /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_jni.h

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
- * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
- * License for the specific language governing permissions and limitations
- * under the License.
- */

Found in path(s):

- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/SocketWritableByteChannel.java
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_limits.c
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_util.c
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_limits.h
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/ErrorsStaticallyReferencedJniMethods.java
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/LimitsStaticallyReferencedJniMethods.java
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/Limits.java
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/io/netty/channel/unix/PeerCredentials.java
- * /opt/cola/permits/1341130306_1654757843.8643026/0/netty-transport-native-unix-common-4-1-72-final-sources-1-jar/netty_unix_util.h

1.181 byte-buddy byte-buddy-1.11.3

1.181.1 Available under license :

ASM: a very small and fast Java bytecode manipulation framework
 Copyright (c) 2000-2011 INRIA, France Telecom
 All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product bundles ASM 9.1, which is available under a "3-clause BSD" license. For details, see licenses/ASM. For more information visit <https://asm.ow2.io>.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2014 - \${current.year} Rafael Winterhalter

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.182 patrickmn-go-cache 2.1.0

1.182.1 Available under license :

This is a list of people who have contributed code to go-cache. They, or their employers, are the copyright holders of the contributed code. Contributed code is subject to the license restrictions listed in LICENSE (as they were when the code was contributed.)

Dustin Sallings <dustin@spy.net>

Jason Mooberry <jasonmoo@me.com>

Sergey Shepelev <temotor@gmail.com>

Alex Edwards <ajmedwards@gmail.com>

Copyright (c) 2012-2017 Patrick Mylund Nielsen and the go-cache contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.183 software-amazon-ion-ion-java 1.0.2

1.183.1 Available under license :

Amazon Ion Java

Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade

names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.184 re2j 1.3

1.184.1 Available under license :

No license file was found, but licenses were detected in source scan.

// Copyright 2010 Google Inc. All Rights Reserved.

Found in path(s):

```
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Pattern.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Matcher.java
```

No license file was found, but licenses were detected in source scan.

```
// Copyright 2010 The Go Authors. All rights reserved.
```

Found in path(s):

```
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/RE2.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Unicode.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Machine.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Parser.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-
jar/com/google/re2j/PatternSyntaxException.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Compiler.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/MachineInput.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Prog.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Inst.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/CharClass.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Utils.java
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Regexp.java
```

No license file was found, but licenses were detected in source scan.

```
// Copyright 2011 The Go Authors. All rights reserved.
```

Found in path(s):

```
* /opt/cola/permits/1168630218_1622042000.35/0/re2j-1-3-sources-3-jar/com/google/re2j/Simplify.java
```

1.185 apache-kafka 2.8.1

1.185.1 Available under license :

Apache Kafka

Copyright 2021 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

This distribution has a binary dependency on jersey, which is available under the CDDL
License. The source code of jersey can be found at <https://github.com/jersey/jersey/>.

The streams-scala (streams/streams-scala) module was donated by Lightbend and the original code was copyrighted
by them:

Copyright (C) 2018 Lightbend Inc. <<https://www.lightbend.com>>

Copyright (C) 2017-2018 Alexis Seigneurin.

This project contains the following code copied from Apache Hadoop:

clients/src/main/java/org/apache/kafka/common/utils/PureJavaCrc32C.java

Some portions of this file Copyright (c) 2004-2006 Intel Corporation and licensed under the BSD license.

This project contains the following code copied from Apache Hive:
streams/src/main/java/org/apache/kafka/streams/state/internals/Murmur3.java

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain

separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the

origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.186 aws-secretsmanager-caching-java 1.0.1

1.186.1 Available under license :

AWS Secrets Manager Java caching client

Copyright 2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation

source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and

may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,

defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.187 golang-protobuf v1.5.2

1.187.1 Available under license :

Copyright 2010 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- * Neither the name of Google Inc. nor the names of its

contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This source code was written by the Go contributors.

The master list of contributors is in the main Go distribution,

visible at <http://tip.golang.org/CONTRIBUTORS>.

1.188 opentest4j-opentest4j 1.2.0

1.188.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,

including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf

of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.189 kotlin 1.3.50

1.189.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
{"version":3,"file":"kotlin.js","sources":["wrapper.js","arrayUtils.js","callableReferenceUtils.js","conversions.js","core.js","long.js","markerFunctions.js","misc.js","polyfills.js","rtti.js","arrayUtils.kt","runtime/Enum.kt","primitiveCompanionObjects.kt","generated/_Arrays.kt","generated/_Ranges.kt","kotlin/UByte.kt","kotlin/UInt.kt","kotlin/UShort.kt","Ranges.kt","collections/Collections.kt","collections/Maps.kt","collections/Sets.kt","text/StringNumberConversions.kt","kotlin/UnsignedUtils.kt","collections/Iterables.kt","collections/Sequences.kt","util/Preconditions.kt","generated/_ArraysJs.kt","comparisons/Comparisons.kt","kotlin/Comparator.kt","util/Standard.kt","generated/_ComparisonsJs.kt","generated/_Collections.kt","kotlin/collections.kt","collections/Iterators.kt","generated/_Comparisons.kt","generated/_Maps.kt","kotlin/ranges.kt","generated/_Sequences.kt","dummy.kt","generated/_Sets.kt","generated/_Strings.kt","text/Strings.kt","kotlin/text/string.kt","kotlin/UByteArray.kt","kotlin/UIntArray.kt","kotlin/ULongArray.kt","kotlin/ULong.kt","kotlin/UShortArray.kt","generated/_UArrays.kt","generated/_UCollections.kt","generated/_
```

UComparisons.kt", "generated/_URanges.kt", "generated/_USequences.kt", "annotation/Annotations.kt", "Collections.kt", "internal/InternalAnnotations.kt", "internal/progressionUtil.kt", "Iterators.kt", "ProgressionIterators.kt", "Progressions.kt", "Range.kt", "reflect/KParameter.kt", "reflect/KType.kt", "reflect/KVariance.kt", "reflect/KVisibility.kt", "Unit.kt", "kotlin/builtins.kt", "kotlin/jsTypeOf.kt", "kotlin/kotlin.kt", "kotlin/coroutines/CoroutineImpl.kt", "util/Result.kt", "coroutines/Continuation.kt", "kotlin/coroutines/intrinsics/IntrinsicsJs.kt", "kotlin/currentBeMisc.kt", "kotlin/jsOperators.kt", "kotlin/math_js-v1.kt", "kotlin/numbers_js-v1.kt", "kotlin/text/numberConversions_js-v1.kt", "generated/_CollectionsJs.kt", "generated/_StringsJs.kt", "generated/_UArraysJs.kt", "jquery/ui.kt", "kotlin/annotations.kt", "kotlin/annotationsJVM.kt", "kotlin/collections/AbstractMutableCollection.kt", "kotlin/collections/AbstractMutableList.kt", "kotlin/collections/AbstractMutableMap.kt", "kotlin/collections/AbstractMutableSet.kt", "kotlin/collections/ArrayList.kt", "kotlin/collections/ArraysJs.kt", "kotlin/collections/ArraySorting.kt", "kotlin/collections/EqualityComparator.kt", "kotlin/collections/HashMap.kt", "kotlin/collections/HashSet.kt", "kotlin/collections/InternalHashMap.kt", "kotlin/collections/InternalMap.kt", "kotlin/collections/InternalStringMap.kt", "kotlin/collections/LinkedHashMap.kt", "kotlin/collections/LinkedHashSet.kt", "kotlin/concurrent.kt", "kotlin/console.kt", "kotlin/coroutines/js/internal/EmptyContinuation.kt", "kotlin/coroutines/SafeContinuationJs.kt", "kotlin/date.kt", "kotlin/dom/Builders.kt", "kotlin/dom/Classes.kt", "text/regex/RegexExtensions.kt", "text/StringBuilder.kt", "kotlin/dom/Dom.kt", "kotlin/dom/EventListener.kt", "kotlin/dom/ItemArrayLike.kt", "kotlin/dom/Mutations.kt", "kotlin/dynamic.kt", "kotlin/exceptions.kt", "kotlin/exceptionUtils.kt", "collections/Grouping.kt", "kotlin/grouping.kt", "kotlin/js.math.kt", "kotlin/json.kt", "kotlin/math.kt", "kotlin/numbers.kt", "kotlin/promise.kt", "kotlin/random/PlatformRandom.kt", "kotlin/reflect/JsClass.kt", "kotlin/reflect/KClassImpl.kt", "kotlin/reflect/primitives.kt", "kotlin/reflect/reflection.kt", "kotlin/regexp.kt", "kotlin/sequence.kt", "kotlin/text/char.kt", "kotlin/text/numberConversions.kt", "kotlin/text/regex.kt", "kotlin/text/stringsCode.kt", "kotlin/text/text.kt", "kotlin/text/utf8Encoding.kt", "kotlin/time/DurationUnit.kt", "kotlin/time/formatToDecimals.kt", "kotlin/time/MonoClock.kt", "org.w3c/org.khronos.webgl.kt", "org.w3c/org.w3c.dom.clipboard.kt", "org.w3c/org.w3c.dom.css.kt", "org.w3c/org.w3c.dom.events.kt", "org.w3c/org.w3c.dom.kt", "org.w3c/org.w3c.fetch.kt", "org.w3c/org.w3c.dom.mediacapture.kt", "org.w3c/org.w3c.dom.pointerevents.kt", "org.w3c/org.w3c.dom.svg.kt", "org.w3c/org.w3c.files.kt", "org.w3c/org.w3c.notifications.kt", "org.w3c/org.w3c.workers.kt", "org.w3c/org.w3c.xhr.kt", "annotations/Experimental.kt", "annotations/ExperimentalStdlibApi.kt", "annotations/Inference.kt", "annotations/Multiplatform.kt", "collections/AbstractCollection.kt", "collections/AbstractIterator.kt", "collections/AbstractList.kt", "collections/AbstractMap.kt", "collections/AbstractSet.kt", "collections/Arrays.kt", "collections/IndexedValue.kt", "collections/MapAccessors.kt", "collections/MapWithDefault.kt", "collections/MutableCollections.kt", "collections/ReversedViews.kt", "collections/SequenceBuilder.kt", "collections/SlidingWindow.kt", "collections/UArraySorting.kt", "contracts/ContractBuilder.kt", "coroutines/ContinuationInterceptor.kt", "coroutines/CoroutineContext.kt", "coroutines/CoroutineContextImpl.kt", "coroutines/intrinsics/Intrinsics.kt", "experimental/bitwiseOperations.kt", "experimental/inferenceMarker.kt", "internal/Annotations.kt", "properties/Delegates.kt", "properties/ObservableProperty.kt", "random/Random.kt", "random/URandom.kt", "random/XorWowRandom.kt", "ranges/Ranges.kt", "reflect/typeOf.kt", "text/Char.kt", "text/Indent.kt", "text/regex/MatchResult.kt", "text/Typography.kt", "time/Clock.kt", "time/Clocks.kt", "time/Duration.kt", "time/DurationUnit.kt", "time/ExperimentalTime.kt", "time/measureTime.kt", "util/HashCode.kt", "util/KotlinVersion.kt", "util/Lateinit.kt", "util/Lazy.kt", "util/Numbers.kt", "util/Suspend.kt", "util/Tuples.kt", "kotlin/annotations/Unsigned.kt", "kotlin/UIntRange.kt", "kotlin/UITerators.kt", "kotlin/ULongRange.kt", "kotlin/UMath.kt", "kotlin/UNumbers.kt", "kotlin/UProgressionUtil.kt", "kotlin/UStrings.kt", "kotlin/MathH.kt", "coroutines.kt", "CoroutinesLibrary.kt", "coroutinesIntrinsics.kt", "ContinuationInterceptor.kt", "CoroutineContext.kt", "CoroutineContextImpl.kt", "Coroutines.kt", "Intrinsics.kt", "SequenceBuilder.kt"], "sourcesContent": ["(function (root, factory) {\r\n if (typeof define === 'function' && define.amd) {\r\n define('kotlin', ['exports'], factory);\r\n }\r\n else if (typeof exports === 'object') {\r\n factory(module.exports);\r\n }\r\n else {\r\n root.kotlin = {};\r\n factory(root.kotlin);\r\n }\r\n}(this, function (Kotlin) {\r\n var _ = Kotlin;\r\n\r\n insertContent();\r\n}));\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\nKotlin.isBooleanArray = function (a) {\r\n return (Array.isArray(a) || a instanceof Int8Array) && a.\$type\$ === \"BooleanArray\"\r\n};\r\n\r\nKotlin.isByteArray = function (a) {\r\n return a instanceof Int8Array && a.\$type\$

```

!=="BooleanArray"\r\n};\r\n\r\nKotlin.isShortArray = function (a) {\r\n    return a instanceof
Int16Array\r\n};\r\n\r\nKotlin.isCharArray = function (a) {\r\n    return a instanceof Uint16Array && a.$type$ ===
"CharArray"\r\n};\r\n\r\nKotlin.isIntArray = function (a) {\r\n    return a instanceof
Int32Array\r\n};\r\n\r\nKotlin.isFloatArray = function (a) {\r\n    return a instanceof
Float32Array\r\n};\r\n\r\nKotlin.isDoubleArray = function (a) {\r\n    return a instanceof
Float64Array\r\n};\r\n\r\nKotlin.isLongArray = function (a) {\r\n    return Array.isArray(a) && a.$type$ ===
"LongArray"\r\n};\r\n\r\nKotlin.isArray = function (a) {\r\n    return Array.isArray(a) &&
!a.$type$;\r\n};\r\n\r\nKotlin.isArrayish = function (a) {\r\n    return Array.isArray(a) ||
ArrayBuffer.isView(a)\r\n};\r\n\r\nKotlin.arrayToString = function (a) {\r\n    var toString = Kotlin.isCharArray(a) ?
String.fromCharCode : Kotlin.toString;\r\n    return "[" + Array.prototype.map.call(a, function(e) { return
toString(e); }).join(", ") + ""]\r\n};\r\n\r\nKotlin.arrayDeepToString = function (arr) {\r\n    return
Kotlin.kotlin.collections.contentDeepToStringImpl(arr);\r\n};\r\n\r\nKotlin.arrayEquals = function (a, b) {\r\n    if (a
=== b) {\r\n        return true;\r\n    }\r\n    if (!Kotlin.isArrayish(b) || a.length !== b.length) {\r\n        return false;\r\n    }
\r\n    for (var i = 0, n = a.length; i < n; i++) {\r\n        if (!Kotlin.equals(a[i], b[i])) {\r\n            return false;\r\n        }
\r\n    }\r\n    return true;\r\n};\r\n\r\nKotlin.arrayDeepEquals = function (a, b) {\r\n    return
Kotlin.kotlin.collections.contentDeepEqualsImpl(a, b);\r\n};\r\n\r\nKotlin.arrayHashCode = function (arr) {\r\n    var
result = 1;\r\n    for (var i = 0, n = arr.length; i < n; i++) {\r\n        result = ((31 * result | 0) +
Kotlin.hashCode(arr[i])) | 0;\r\n    }\r\n    return result;\r\n};\r\n\r\nKotlin.arrayDeepHashCode = function (arr) {\r\n
return Kotlin.kotlin.collections.contentDeepHashCodeImpl(arr);\r\n};\r\n\r\nKotlin.primitiveArraySort = function
(array) {\r\n    array.sort(Kotlin.doubleCompareTo);\r\n};\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and
Kotlin Programming Language contributors. \r\n * Use of this source code is governed by the Apache 2.0 license
that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\nKotlin.getCallableRef = function(name, f) {\r\n
f.callableName = name;\r\n    return f;\r\n};\r\n\r\nKotlin.getPropertyCallableRef = function(name, paramCount,
getter, setter) {\r\n    getter.get = getter;\r\n    getter.set = setter;\r\n    getter.callableName = name;\r\n    return
getPropertyRefClass(getter, setter, propertyRefClassMetadataCache[paramCount]);\r\n};\r\n\r\n\r\nfunction
getPropertyRefClass(obj, setter, cache) {\r\n    obj.$metadata$ = getPropertyRefMetadata(typeof setter ===
"function" ? cache.mutable : cache.immutable);\r\n    obj.constructor = obj;\r\n    return obj;\r\n}\r\n\r\nvar
propertyRefClassMetadataCache = [\r\n    {\r\n        mutable: { value: null, implementedInterface: function () {\r\n
return Kotlin.kotlin.reflect.KMutableProperty0 }\r\n        },\r\n        immutable: { value: null,
implementedInterface: function () {\r\n            return Kotlin.kotlin.reflect.KProperty0 }\r\n        }
\r\n    },\r\n    {\r\n        mutable: { value: null, implementedInterface: function () {\r\n            return
Kotlin.kotlin.reflect.KMutableProperty1 }\r\n        },\r\n        immutable: { value: null, implementedInterface:
function () {\r\n            return Kotlin.kotlin.reflect.KProperty1 }\r\n        }
\r\n    }
\r\n];\r\n\r\nfunction
getPropertyRefMetadata(cache) {\r\n    if (cache.value === null) {\r\n        cache.value = {\r\n            interfaces:
[cache.implementedInterface()],\r\n            baseClass: null,\r\n            functions: {},\r\n            properties: {},\r\n
            types: {},\r\n            staticMembers: {}
\r\n        };\r\n    }\r\n    return cache.value;\r\n}\r\n", "/*\r\n * Copyright
2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \r\n * Use of this source code is
governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\nKotlin.toShort =
function (a) {\r\n    return (a & 0xFFFF) << 16 >> 16;\r\n};\r\n\r\nKotlin.toByte = function (a) {\r\n    return (a &
0xFF) << 24 >> 24;\r\n};\r\n\r\nKotlin.toChar = function (a) {\r\n    return a &
0xFFFF;\r\n};\r\n\r\nKotlin.numberToLong = function (a) {\r\n    return a instanceof Kotlin.Long ? a :
Kotlin.Long.fromNumber(a);\r\n};\r\n\r\nKotlin.numberToInt = function (a) {\r\n    return a instanceof Kotlin.Long
? a.toInt() : Kotlin.doubleToInt(a);\r\n};\r\n\r\nKotlin.numberToShort = function (a) {\r\n    return
Kotlin.toShort(Kotlin.numberToInt(a));\r\n};\r\n\r\nKotlin.numberToByte = function (a) {\r\n    return
Kotlin.toByte(Kotlin.numberToInt(a));\r\n};\r\n\r\nKotlin.numberToDouble = function (a) {\r\n    return
+a;\r\n};\r\n\r\nKotlin.numberToChar = function (a) {\r\n    return
Kotlin.toChar(Kotlin.numberToInt(a));\r\n};\r\n\r\nKotlin.doubleToInt = function(a) {\r\n    if (a > 2147483647)
return 2147483647;\r\n    if (a < -2147483648) return -2147483648;\r\n    return a |

```

```

0;};\n\nKotlin.toBoxedChar = function (a) {\n  if (a == null) return a;\n  if (a instanceof\nKotlin.BoxedChar) return a;\n  return new Kotlin.BoxedChar(a);};\n\nKotlin.unboxChar = function(a) {\n  if (a == null) return a;\n  return Kotlin.toChar(a);};\n\n"/**\n * Copyright 2010-2018 JetBrains s.r.o. and\nKotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license\nthat can be found in the license/LICENSE.txt file.\n */\n\nKotlin.equals = function (obj1, obj2) {\n  if (obj1\n== null) {\n    return obj2 == null;\n  }\n  if (obj2 == null) {\n    return false;\n  }\n  if\n(obj1 !== obj2) {\n    return obj2 !== obj2;\n  }\n  if (typeof obj1 === "object" && typeof obj1.equals\n=== "function") {\n    return obj1.equals(obj2);\n  }\n  if (typeof obj1 === "number" && typeof obj2\n=== "number") {\n    return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);\n  }\n  return obj1\n=== obj2;};\n\nKotlin.hashCode = function (obj) {\n  if (obj == null) {\n    return 0;\n  }\n  var\nobjType = typeof obj;\n  if ("object" === objType) {\n    return "function" === typeof obj.hashCode ?\nobj.hashCode() : getObjectHashCode(obj);}\n  if ("function" === objType) {\n    return\ngetObjectHashCode(obj);}\n  if ("number" === objType) {\n    return\nKotlin.numberHashCode(obj);}\n  if ("boolean" === objType) {\n    return Number(obj)}\n  }\n  var str = String(obj);\n  return getStringHashCode(str);};\n\nKotlin.toString = function (o) {\n  if (o\n== null) {\n    return "null";\n  }\n  else if (Kotlin.isArrayish(o)) {\n    return "[...]";\n  }\n  else\n{\n    return o.toString();\n  }\n};\n\n** @const *\nvar POW_2_32 = 4294967296;\n\n// TODO:\nconsider switching to Symbol type once we are on ES6.\n** @const *\nvar\nOBJECT_HASH_CODE_PROPERTY_NAME = "kotlinHashCodeValue";\n\nfunction\ngetObjectHashCode(obj) {\n  if (!(OBJECT_HASH_CODE_PROPERTY_NAME in obj)) {\n    var hash =\n(Math.random() * POW_2_32) | 0; // Make 32-bit signed integer.\n    Object.defineProperty(obj,\nOBJECT_HASH_CODE_PROPERTY_NAME, { value: hash, enumerable: false });\n  }\n  return\nobj[OBJECT_HASH_CODE_PROPERTY_NAME];\n}\n\nfunction getStringHashCode(str) {\n  var hash =\n0;\n  for (var i = 0; i < str.length; i++) {\n    var code = str.charCodeAt(i);\n    hash = (hash * 31 + code) |\n0; // Keep it 32-bit.\n  }\n  return hash;\n}\n\nKotlin.identityHashCode = getObjectHashCode;\n\n"/**\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source\ncode is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\nCopyright 2009 The Closure Library Authors. All Rights Reserved.\n\nLicensed under the Apache License,\nVersion 2.0 (the "License");\n\nyou may not use this file except in compliance with the License.\n\nYou may\nobtain a copy of the License at\n\nhttp://www.apache.org/licenses/LICENSE-2.0\n\nUnless required\nby applicable law or agreed to in writing, software\ndistributed under the License is distributed on an "AS-IS"\nBASIS,\n\nWITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or\nimplied.\n\n**\n\n * Constructs a 64-bit two's-complement integer, given its low and high 32-bit\n * values as\n * signed integers. See the from* functions below for more\n * convenient ways of constructing Longs.\n\n * The internal representation of a long is the two given signed, 32-bit values. We use 32-bit pieces because these\n * are the size of integers on which Javascript performs bit-operations. For operations like addition and\n * multiplication, we split each number into 16-bit pieces, which can easily be multiplied within Javascript's\n * floating-point representation without overflow or change in sign.\n\n * In the algorithms below, we frequently reduce the negative case to the positive case by negating the input(s) and then post-processing the\n * result. Note that we must ALWAYS check specially whether those values are MIN_VALUE (-2^63)\n * because -MIN_VALUE == MIN_VALUE (since 2^63 cannot be represented as a positive number, it\n * overflows back into a negative). Not handling this case would often result in infinite recursion.\n\n * @param {number} low The low (signed) 32 bits of the long.\n * @param {number} high The high (signed) 32\n * bits of the long.\n * @constructor\n * @final\n */\nKotlin.Long = function(low, high) {\n  /**\n * @type\n * {number}\n * @private\n */\n  this.low_ = low | 0; // force into 32 signed bits.\n\n  /**\n * @type\n * {number}\n * @private\n */\n  this.high_ = high | 0; // force into 32 signed\n * bits.\n};\n\nKotlin.Long.$metadata$ = {\n  kind: "class",\n  simpleName: "Long",\n  interfaces: []\n};\n\n// NOTE: Common constant values ZERO, ONE, NEG_ONE, etc. are defined below

```

```

the from* methods on which they depend.
A cache of the Long representations of small
integer values.
 * @type {!Object}
 * @private
 * Kotlin.Long.IntCache_ = {};
 * Returns a Long representing the given (32-bit) integer value.
 * @param {number} value The 32-bit integer in
question.
 * @return {!Kotlin.Long} The corresponding Long value.
 * Kotlin.Long.fromInt =
function(value) {
    if (-128 <= value && value < 128) {
        var cachedObj = Kotlin.Long.IntCache_[value];
        if (cachedObj) {
            return cachedObj;
        }
        var obj = new Kotlin.Long(value | 0, value < 0 ? -1 :
0);
        if (-128 <= value && value < 128) {
            Kotlin.Long.IntCache_[value] = obj;
        }
        return
obj;
    };
 * Converts this number value to `Long`.
 * The fractional part, if any, is rounded
down.
 * Returns zero if this `Double` value is `NaN`, `Long.MIN_VALUE` if it's less than
`Long.MIN_VALUE`, `Long.MAX_VALUE` if it's bigger than `Long.MAX_VALUE`.
 * @param
{number} value The number in question.
 * @return {!Kotlin.Long} The corresponding Long value.
 * Kotlin.Long.fromNumber = function(value) {
    if (isNaN(value)) {
        return Kotlin.Long.ZERO;
    }
    else if (value <= -Kotlin.Long.TWO_PWR_63_DBL_) {
        return Kotlin.Long.MIN_VALUE;
    }
    else if (value + 1 >= Kotlin.Long.TWO_PWR_63_DBL_) {
        return Kotlin.Long.MAX_VALUE;
    }
    else if (value <
0) {
        return Kotlin.Long.fromNumber(-value).negate();
    }
    else {
        return new Kotlin.Long(
(value % Kotlin.Long.TWO_PWR_32_DBL_) | 0,
        (value / Kotlin.Long.TWO_PWR_32_DBL_) | 0);
    };
};
 * Returns a Long representing the 64-bit integer that comes by concatenating
the given
high and low bits. Each is assumed to use 32 bits.
 * @param {number} lowBits The low 32-bits.
 * @param
{number} highBits The high 32-bits.
 * @return {!Kotlin.Long} The corresponding Long value.
 * Kotlin.Long.fromBits = function(lowBits, highBits) {
    return new Kotlin.Long(lowBits,
highBits);
};
 * Returns a Long representation of the given string, written using the given
 * radix.
 * @param {string} str The textual representation of the Long.
 * @param {number=} opt_radix The
radix in which the text is written.
 * @return {!Kotlin.Long} The corresponding Long value.
 * Kotlin.Long.fromString = function(str, opt_radix) {
    if (str.length == 0) {
        throw Error('number format
error: empty string');
    }
    var radix = opt_radix || 10;
    if (radix < 2 || 36 < radix) {
        throw
Error('radix out of range: ' + radix);
    }
    if (str.charAt(0) == '-') {
        return
Kotlin.Long.fromString(str.substring(1), radix).negate();
    }
    else if (str.indexOf('-') >= 0) {
        throw
Error('number format error: interior "-" character: ' + str);
    }
    // Do several (8) digits each time through
the loop, so as to
    // minimize the calls to the very expensive emulated div.
    var radixToPower =
Kotlin.Long.fromNumber(Math.pow(radix, 8));
    var result = Kotlin.Long.ZERO;
    for (var i = 0; i <
str.length; i += 8) {
        var size = Math.min(8, str.length - i);
        var value = parseInt(str.substring(i, i + size),
radix);
        if (size < 8) {
            var power = Kotlin.Long.fromNumber(Math.pow(radix, size));
            result =
result.multiply(power).add(Kotlin.Long.fromNumber(value));
        }
        else {
            result =
result.multiply(radixToPower);
            result = result.add(Kotlin.Long.fromNumber(value));
        }
    }
    return
result;
};
// NOTE: the compiler should inline these constant values below and then remove
// these
variables, so there should be no runtime penalty for these.
 * Number used repeated below in
calculations. This must appear before the
 * first call to any from* function below.
 * @type {number}
 * @private
 * Kotlin.Long.TWO_PWR_16_DBL_ = 1 << 16;
 * @type {number}
 * @private
 * Kotlin.Long.TWO_PWR_24_DBL_ = 1 << 24;
 * @type {number}
 * @private
 * Kotlin.Long.TWO_PWR_32_DBL_ =
Kotlin.Long.TWO_PWR_16_DBL_ *
Kotlin.Long.TWO_PWR_16_DBL_;
 * @type {number}
 * @private
 * Kotlin.Long.TWO_PWR_31_DBL_ =
Kotlin.Long.TWO_PWR_32_DBL_ / 2;
 * @type
{number}
 * @private
 * Kotlin.Long.TWO_PWR_48_DBL_ =
Kotlin.Long.TWO_PWR_32_DBL_
 * Kotlin.Long.TWO_PWR_16_DBL_;
 * @type {number}
 * @private
 * Kotlin.Long.TWO_PWR_64_DBL_ =
Kotlin.Long.TWO_PWR_32_DBL_ *
Kotlin.Long.TWO_PWR_32_DBL_;
 * @type {number}
 * @private
 * Kotlin.Long.TWO_PWR_63_DBL_ =
Kotlin.Long.TWO_PWR_64_DBL_ / 2;
 * @type
{!Kotlin.Long}
 * Kotlin.Long.ZERO = Kotlin.Long.fromInt(0);
 * @type {!Kotlin.Long}

```

```

* Kotlin.Long.ONE = Kotlin.Long.fromInt(1);
* Kotlin.Long.NEG_ONE = Kotlin.Long.fromInt(-1);
* Kotlin.Long.MAX_VALUE = Kotlin.Long.fromBits(0xFFFFFFFF | 0, 0x7FFFFFFF | 0);
* Kotlin.Long.MIN_VALUE = Kotlin.Long.fromBits(0, 0x80000000 | 0);
* Kotlin.Long.TWO_PWR_24_ = Kotlin.Long.fromInt(1 << 24);
* Kotlin.Long.prototype.toInt = function() { return this.low_; }
* Kotlin.Long.prototype.toNumber = function() { return this.high_ * Kotlin.Long.TWO_PWR_32_DBL_ + this.getLowBitsUnsigned(); }
* Kotlin.Long.prototype.hashCode = function() { return this.high_ ^ this.low_; }
* Kotlin.Long.prototype.toString = function(opt_radix) {
    var radix = opt_radix || 10;
    if (radix < 2 || 36 < radix) throw Error('radix out of range: ' + radix);
    if (this.isZero()) return '0';
    if (this.isNegative()) {
        if (this.equalsLong(Kotlin.Long.MIN_VALUE)) // We need to change the Long value before it can be negated, so we remove the bottom-most digit in this base and then recurse to do the rest.
            var radixLong = Kotlin.Long.fromNumber(radix);
            var div = this.div(radixLong);
            var rem = div.multiply(radixLong).subtract(this);
            return div.toString(radix) + rem.toInt().toString(radix);
        } else {
            return '-' + this.negate().toString(radix);
        }
    }
    // Do several (6) digits each time through the loop, so as to minimize the calls to the very expensive emulated div.
    var radixToPower = Kotlin.Long.fromNumber(Math.pow(radix, 6));
    var rem = this;
    var result = '';
    while (true) {
        var remDiv = rem.div(radixToPower);
        var intVal = rem.subtract(remDiv.multiply(radixToPower)).toInt();
        var digits = intVal.toString(radix);
        rem = remDiv;
        if (rem.isZero()) return digits + result;
        else {
            while (digits.length < 6) digits = '0' + digits;
            result = '' + digits + result;
        }
    }
}
* Kotlin.Long.prototype.getHighBits = function() { return this.high_; }
* Kotlin.Long.prototype.getLowBits = function() { return this.low_; }
* Kotlin.Long.prototype.getLowBitsUnsigned = function() { return (this.low_ >= 0) ? this.low_ : Kotlin.Long.TWO_PWR_32_DBL_ + this.low_; }
* Kotlin.Long.prototype.getNumBitsAbs = function() {
    if (this.isNegative()) {
        if (this.equalsLong(Kotlin.Long.MIN_VALUE)) return 64;
    } else {
        return this.negate().getNumBitsAbs();
    }
    var val = this.high_ != 0 ? this.high_ : this.low_;
    for (var bit = 31; bit > 0; bit--) {
        if ((val & (1 << bit)) != 0) break;
    }
    return this.high_ != 0 ? bit + 33 : bit + 1;
}
* Kotlin.Long.prototype.isZero = function() { return this.high_ == 0 && this.low_ == 0; }
* Kotlin.Long.prototype.isNegative = function() { return this.high_ < 0; }
* Kotlin.Long.prototype.isOdd = function() { return (this.low_ & 1) == 1; }
* Kotlin.Long.prototype.equalsLong = function(other) { return (this.high_ == other.high_) && (this.low_ == other.low_); }
* Kotlin.Long.prototype.notEqualsLong = function(other) { return (this.high_ != other.high_) || (this.low_ != other.low_); }
* Kotlin.Long.prototype.lessThan = function(other) { return this.compare(other) < 0; }
* Kotlin.Long.prototype.lessThanOrEqual = function(other) { return

```



```
(other.isZero()) {\r\n    throw Error('division by zero');\r\n } else if (this.isZero()) {\r\n    return\r\n    Kotlin.Long.ZERO;\r\n }\r\n\r\n if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {\r\n    if\r\n    (other.equalsLong(Kotlin.Long.ONE) ||\r\n     other.equalsLong(Kotlin.Long.NEG_ONE)) {\r\n    return\r\n    Kotlin.Long.MIN_VALUE; // recall that -MIN_VALUE == MIN_VALUE\r\n } else if\r\n (other.equalsLong(Kotlin.Long.MIN_VALUE)) {\r\n    return Kotlin.Long.ONE;\r\n } else {\r\n // At this\r\n point, we have |other| >= 2, so |this/other| < |MIN_VALUE|.\r\n    var halfThis = this.shiftRight(1);\r\n    var\r\n    approx = halfThis.div(other).shiftLeft(1);\r\n    if (approx.equalsLong(Kotlin.Long.ZERO)) {\r\n    return\r\n    other.isNegative() ? Kotlin.Long.ONE : Kotlin.Long.NEG_ONE;\r\n } else {\r\n    var rem =\r\n    this.subtract(other.multiply(approx));\r\n    var result = approx.add(rem.div(other));\r\n    return result;\r\n }\r\n }\r\n } else if (other.equalsLong(Kotlin.Long.MIN_VALUE)) {\r\n    return Kotlin.Long.ZERO;\r\n }\r\n\r\n if (this.isNegative()) {\r\n    if (other.isNegative()) {\r\n    return this.negate().div(other.negate());\r\n } else {\r\n    return this.negate().div(other).negate();\r\n }\r\n }\r\n } else if (other.isNegative()) {\r\n    return\r\n    this.div(other.negate()).negate();\r\n }\r\n\r\n // Repeat the following until the remainder is less than other: find\r\n a\r\n // floating-point that approximates remainder / other *from below*, add this\r\n // into the result, and subtract\r\n it from the remainder. It is critical that\r\n // the approximate value is less than or equal to the real value so that\r\n the\r\n // remainder never becomes negative.\r\n    var res = Kotlin.Long.ZERO;\r\n    var rem = this;\r\n    while\r\n (rem.greaterThanOrEqual(other)) {\r\n // Approximate the result of division. This may be a little greater or\r\n // smaller than the actual value.\r\n    var approx = Math.max(1, Math.floor(rem.toNumber() /\r\n    other.toNumber()));\r\n\r\n // We will tweak the approximate result by changing it in the 48-th digit or\r\n // the\r\n // smallest non-fractional digit, whichever is larger.\r\n    var log2 = Math.ceil(Math.log(approx) / Math.LN2);\r\n    var delta = (log2 <= 48) ? 1 : Math.pow(2, log2 - 48);\r\n\r\n // Decrease the approximation until it is smaller than\r\n the remainder. Note\r\n // that if it is too large, the product overflows and is negative.\r\n    var approxRes =\r\n    Kotlin.Long.fromNumber(approx);\r\n    var approxRem = approxRes.multiply(other);\r\n    while\r\n (approxRem.isNegative() || approxRem.greaterThan(rem)) {\r\n    approx -= delta;\r\n    approxRes =\r\n    Kotlin.Long.fromNumber(approx);\r\n    approxRem = approxRes.multiply(other);\r\n }\r\n }\r\n // We know the\r\n answer can't be zero... and actually, zero would cause\r\n // infinite recursion since we would make no\r\n progress.\r\n    if (approxRes.isZero()) {\r\n    approxRes = Kotlin.Long.ONE;\r\n }\r\n\r\n res =\r\n res.add(approxRes);\r\n    rem = rem.subtract(approxRem);\r\n }\r\n    return res;\r\n};\r\n\r\n /**\r\n * Returns this\r\n Long modulo the given one.\r\n * @param {Kotlin.Long} other Long by which to mod.\r\n * @return\r\n {!Kotlin.Long} This Long modulo the given one.\r\n */\r\nKotlin.Long.prototype.modulo = function(other) {\r\n    return this.subtract(this.div(other).multiply(other));\r\n};\r\n\r\n /**\r\n * @return {!Kotlin.Long} The bitwise-NOT of\r\n this value.\r\n */\r\nKotlin.Long.prototype.not = function() {\r\n    return Kotlin.Long.fromBits(~this.low_,\r\n ~this.high_);\r\n};\r\n\r\n /**\r\n * Returns the bitwise-AND of this Long and the given one.\r\n * @param\r\n {Kotlin.Long} other The Long with which to AND.\r\n * @return {!Kotlin.Long} The bitwise-AND of this and the\r\n other.\r\n */\r\nKotlin.Long.prototype.and = function(other) {\r\n    return Kotlin.Long.fromBits(this.low_ &\r\n    other.low_,\r\n    this.high_ & other.high_);\r\n};\r\n\r\n /**\r\n * Returns the bitwise-OR of\r\n this Long and the given one.\r\n * @param {Kotlin.Long} other The Long with which to OR.\r\n * @return\r\n {!Kotlin.Long} The bitwise-OR of this and the other.\r\n */\r\nKotlin.Long.prototype.or = function(other) {\r\n    return Kotlin.Long.fromBits(this.low_ |\r\n    other.low_ |,\r\n    this.high_ |,\r\n    other.high_);\r\n};\r\n\r\n /**\r\n * Returns the bitwise-XOR of this Long and the given one.\r\n * @param\r\n {Kotlin.Long} other The Long with which to XOR.\r\n * @return {!Kotlin.Long} The bitwise-XOR of this and the\r\n other.\r\n */\r\nKotlin.Long.prototype.xor = function(other) {\r\n    return Kotlin.Long.fromBits(this.low_ ^\r\n    other.low_,\r\n    this.high_ ^ other.high_);\r\n};\r\n\r\n /**\r\n * Returns this Long with bits\r\n shifted to the left by the given amount.\r\n * @param {number} numBits The number of bits by which to shift.\r\n * @return {!Kotlin.Long} This shifted to the left by the given amount.\r\n */\r\nKotlin.Long.prototype.shiftLeft =\r\n function(numBits) {\r\n    numBits &= 63;\r\n    if (numBits == 0) {\r\n    return this;\r\n } else {\r\n    var low =\r\n    this.low_;\r\n    if (numBits < 32) {\r\n    var high = this.high_;\r\n    return Kotlin.Long.fromBits((low << numBits,\r\n    (high << numBits) | (low >>> (32 - numBits)));\r\n } else {\r\n    return
```

```

Kotlin.Long.fromBits(0, low << (numBits - 32));\r\n } \r\n } \r\n};\r\n\r\n\r\n\r\n**\r\n * Returns this Long with bits
shifted to the right by the given amount.\r\n * @param {number} numBits The number of bits by which to shift.\r\n
* @return {!Kotlin.Long} This shifted to the right by the given amount.\r\n */\r\nKotlin.Long.prototype.shiftRight =
function(numBits) {\r\n numBits &= 63;\r\n if (numBits == 0) {\r\n return this;\r\n } else {\r\n var high =
this.high_;\r\n if (numBits < 32) {\r\n var low = this.low_;\r\n return Kotlin.Long.fromBits(\r\n (low
>>> numBits) | (high << (32 - numBits)),\r\n high >> numBits);\r\n } else {\r\n return
Kotlin.Long.fromBits(\r\n high >> (numBits - 32),\r\n high >= 0 ? 0 : -1);\r\n } \r\n
}\r\n};\r\n\r\n\r\n\r\n**\r\n * Returns this Long with bits shifted to the right by the given amount, with\r\n * zeros
placed into the new leading bits.\r\n * @param {number} numBits The number of bits by which to shift.\r\n *
@return {!Kotlin.Long} This shifted to the right by the given amount, with\r\n * zeros placed into the new
leading bits.\r\n */\r\nKotlin.Long.prototype.shiftRightUnsigned = function(numBits) {\r\n numBits &= 63;\r\n if
(numBits == 0) {\r\n return this;\r\n } else {\r\n var high = this.high_;\r\n if (numBits < 32) {\r\n var low =
this.low_;\r\n return Kotlin.Long.fromBits(\r\n (low >>> numBits) | (high << (32 - numBits)),\r\n high
>>> numBits);\r\n } else if (numBits == 32) {\r\n return Kotlin.Long.fromBits(high, 0);\r\n } else {\r\n
return Kotlin.Long.fromBits(high >>> (numBits - 32), 0);\r\n } \r\n } \r\n};\r\n\r\n\r\n\r\n// Support for
Kotlin\r\nKotlin.Long.prototype.equals = function (other) {\r\n return other instanceof Kotlin.Long &&
this.equalsLong(other);\r\n};\r\n\r\n\r\nKotlin.Long.prototype.compareTo_11rb$ =
Kotlin.Long.prototype.compare;\r\n\r\n\r\nKotlin.Long.prototype.inc = function() {\r\n return
this.add(Kotlin.Long.ONE);\r\n};\r\n\r\n\r\nKotlin.Long.prototype.dec = function() {\r\n return
this.add(Kotlin.Long.NEG_ONE);\r\n};\r\n\r\n\r\nKotlin.Long.prototype.valueOf = function() {\r\n return
this.toNumber();\r\n};\r\n\r\n\r\nKotlin.Long.prototype.unaryPlus = function() {\r\n return
this;\r\n};\r\n\r\n\r\nKotlin.Long.prototype.unaryMinus = Kotlin.Long.prototype.negate;\r\n\r\n\r\nKotlin.Long.prototype.inv =
Kotlin.Long.prototype.not;\r\n\r\n\r\nKotlin.Long.prototype.rangeTo = function (other) {\r\n return new
Kotlin.kotlin.ranges.LongRange(this, other);\r\n};\r\n\r\n\r\n\r\n**\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin
Programming Language contributors. \r\n * Use of this source code is governed by the Apache 2.0 license that can
be found in the license/LICENSE.txt file.\r\n */\r\n\r\n\r\n\r\n**\r\n * @param {string} id\r\n * @param {Object}
declaration\r\n */\r\nKotlin.defineModule = function (id, declaration) {\r\n};\r\n\r\n\r\n\r\nKotlin.defineInlineFunction =
function(tag, fun) {\r\n return fun;\r\n};\r\n\r\n\r\n\r\nKotlin.wrapFunction = function(fun) {\r\n var f = function() {\r\n
f = fun();\r\n return f.apply(this, arguments);\r\n }; \r\n return function() {\r\n return f.apply(this,
arguments);\r\n }; \r\n};\r\n\r\n\r\n\r\nKotlin.isTypeOf = function(type) {\r\n return function (object) {\r\n return
typeof object === type;\r\n } \r\n};\r\n\r\n\r\n\r\nKotlin.isInstanceOf = function (klass) {\r\n return function (object)
{\r\n return Kotlin.isType(object, klass);\r\n } \r\n};\r\n\r\n\r\n\r\nKotlin.orNull = function (fn) {\r\n return function
(object) {\r\n return object == null || fn(object);\r\n } \r\n};\r\n\r\n\r\n\r\nKotlin.andPredicate = function (a, b) {\r\n
return function (object) {\r\n return a(object) && b(object);\r\n } \r\n};\r\n\r\n\r\n\r\nKotlin.kotlinModuleMetadata =
function (abiVersion, moduleName, data) {\r\n};\r\n\r\n\r\n\r\nKotlin.suspendCall = function(value) {\r\n return
value;\r\n};\r\n\r\n\r\n\r\nKotlin.coroutineResult = function(qualifier) {\r\n
throwMarkerError();\r\n};\r\n\r\n\r\n\r\nKotlin.coroutineController = function(qualifier) {\r\n
throwMarkerError();\r\n};\r\n\r\n\r\n\r\nKotlin.coroutineReceiver = function(qualifier) {\r\n
throwMarkerError();\r\n};\r\n\r\n\r\n\r\nKotlin.setCoroutineResult = function(value, qualifier) {\r\n
throwMarkerError();\r\n};\r\n\r\n\r\n\r\nfunction throwMarkerError() {\r\n throw new Error(\r\n \r\n "This marker
function should never be called. \r\n" +\r\n \r\n "Looks like compiler did not eliminate it properly. \r\n" +\r\n
\r\n "Please, report an issue if you caught this exception.");\r\n};\r\n\r\n\r\n\r\n\r\nKotlin.getFunctionById = function(id,
defaultValue) {\r\n return function() {\r\n return defaultValue;\r\n } \r\n};\r\n\r\n\r\n\r\n\r\n**\r\n * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors. \r\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n\r\n\r\n\r\nKotlin.compareTo = function (a,
b) {\r\n var typeA = typeof a;\r\n if (typeA === "number") {\r\n if (typeof b === "number") {\r\n
return Kotlin.doubleCompareTo(a, b);\r\n } \r\n return Kotlin.primitiveCompareTo(a, b);\r\n } \r\n if
(typeA === "string" || typeA === "boolean") {\r\n return Kotlin.primitiveCompareTo(a, b);\r\n } \r\n
}\r\n};

```

```

return a.compareTo_11rb$(b);\r\n};\r\n\r\nKotlin.primitiveCompareTo = function (a, b) {\r\n  return a < b ? -1 : a >
b ? 1 : 0;\r\n};\r\n\r\nKotlin.doubleCompareTo = function (a, b) {\r\n  if (a < b) return -1;\r\n  if (a > b) return
1;\r\n\r\n  if (a === b) {\r\n    if (a !== 0) return 0;\r\n\r\n    var ia = 1 / a;\r\n    return ia === 1 / b ? 0 : (ia <
0 ? -1 : 1);\r\n  }\r\n\r\n  return a !== a ? (b !== b ? 0 : 1) : -1\r\n};\r\n\r\nKotlin.charInc = function (value) {\r\n
return Kotlin.toChar(value+1);\r\n};\r\n\r\nKotlin.charDec = function (value) {\r\n  return Kotlin.toChar(value-
1);\r\n};\r\n\r\nKotlin.imul = Math.imul || imul;\r\n\r\nKotlin.imulEmulated = imul;\r\n\r\nfunction imul(a, b) {\r\n
return ((a & 0xffff0000) * (b & 0xffff) + (a & 0xffff) * (b | 0)) | 0;\r\n}\r\n\r\n(function() {\r\n  var buf = new
ArrayBuffer(8);\r\n  var bufFloat64 = new Float64Array(buf);\r\n  var bufFloat32 = new Float32Array(buf);\r\n
var bufInt32 = new Int32Array(buf);\r\n  var lowIndex = 0;\r\n  var highIndex = 1;\r\n\r\n  bufFloat64[0] = -1; //
bff00000_00000000\r\n  if (bufInt32[lowIndex] !== 0) {\r\n    lowIndex = 1;\r\n    highIndex = 0;\r\n
}\r\n\r\n  Kotlin.doubleToBits = function(value) {\r\n    return Kotlin.doubleToRawBits(isNaN(value) ? NaN :
value);\r\n  };\r\n\r\n  Kotlin.doubleToRawBits = function(value) {\r\n    bufFloat64[0] = value;\r\n    return
Kotlin.Long.fromBits(bufInt32[lowIndex], bufInt32[highIndex]);\r\n  };\r\n\r\n  Kotlin.doubleFromBits =
function(value) {\r\n    bufInt32[lowIndex] = value.low_;\r\n    bufInt32[highIndex] = value.high_;\r\n
return bufFloat64[0];\r\n  };\r\n\r\n  Kotlin.floatToBits = function(value) {\r\n    return
Kotlin.floatToRawBits(isNaN(value) ? NaN : value);\r\n  };\r\n\r\n  Kotlin.floatToRawBits = function(value) {\r\n
  bufFloat32[0] = value;\r\n    return bufInt32[0];\r\n  };\r\n\r\n  Kotlin.floatFromBits = function(value) {\r\n
  bufInt32[0] = value;\r\n    return bufFloat32[0];\r\n  };\r\n\r\n  // returns zero value for number with positive
sign bit and non-zero value for number with negative sign bit.\r\n  Kotlin.doubleSignBit = function(value) {\r\n
bufFloat64[0] = value;\r\n    return bufInt32[highIndex] & 0x80000000;\r\n  };\r\n\r\n  Kotlin.numberHashCode
= function(obj) {\r\n    if ((obj | 0) === obj) {\r\n      return obj | 0;\r\n    }\r\n    else {\r\n
bufFloat64[0] = obj;\r\n      return (bufInt32[highIndex] * 31 | 0) + bufInt32[lowIndex] | 0;\r\n    }\r\n
}\r\n\r\n})());\r\n\r\nKotlin.ensureNotNull = function(x) {\r\n  return x != null ? x :
Kotlin.throwNPE();\r\n};\r\n\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors. \r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\nif (typeof String.prototype.startsWith === "undefined") {\r\n
String.prototype.startsWith = function(searchString, position) {\r\n  position = position || 0;\r\n  return
this.lastIndexOf(searchString, position) === position;\r\n  };\r\n}\r\n\r\nif (typeof String.prototype.endsWith ===
"undefined") {\r\n  String.prototype.endsWith = function(searchString, position) {\r\n    var subjectString =
this.toString();\r\n    if (position === undefined || position > subjectString.length) {\r\n      position =
subjectString.length;\r\n    }\r\n    position -= searchString.length;\r\n    var lastIndex =
subjectString.indexOf(searchString, position);\r\n    return lastIndex !== -1 && lastIndex === position;\r\n
};\r\n}\r\n\r\n// ES6 Math polyfills\r\n\r\nif (typeof Math.sign === "undefined") {\r\n  Math.sign = function(x) {\r\n
x = +x; // convert to a number\r\n  if (x === 0 || isNaN(x)) {\r\n    return Number(x);\r\n  }\r\n
return x > 0 ? 1 : -1;\r\n  };\r\n}\r\n\r\nif (typeof Math.trunc === "undefined") {\r\n  Math.trunc = function(x) {\r\n
if (isNaN(x)) {\r\n    return NaN;\r\n  }\r\n  if (x > 0) {\r\n    return Math.floor(x);\r\n  }\r\n
return Math.ceil(x);\r\n  };\r\n}\r\n\r\n(function() {\r\n  var epsilon = 2.220446049250313E-16;\r\n  var
taylor_2_bound = Math.sqrt(epsilon);\r\n  var taylor_n_bound = Math.sqrt(taylor_2_bound);\r\n  var
upper_taylor_2_bound = 1/taylor_2_bound;\r\n  var upper_taylor_n_bound = 1/taylor_n_bound;\r\n\r\n  if (typeof
Math.sinh === "undefined") {\r\n    Math.sinh = function(x) {\r\n      if (Math.abs(x) < taylor_n_bound) {\r\n
var result = x;\r\n      if (Math.abs(x) > taylor_2_bound) {\r\n        result += (x * x * x) / 6;\r\n
}\r\n      return result;\r\n    } else {\r\n      var y = Math.exp(x);\r\n      var y1 = 1 /
y;\r\n      if (!isFinite(y)) return Math.exp(x - Math.LN2);\r\n      if (!isFinite(y1)) return -Math.exp(-x -
Math.LN2);\r\n      return (y - y1) / 2;\r\n    }\r\n  };\r\n}\r\n\r\n  if (typeof Math.cosh ===
"undefined") {\r\n    Math.cosh = function(x) {\r\n      var y = Math.exp(x);\r\n      var y1 = 1 / y;\r\n
if (!isFinite(y) || !isFinite(y1)) return Math.exp(Math.abs(x) - Math.LN2);\r\n      return (y + y1) / 2;\r\n
};\r\n}\r\n\r\n  if (typeof Math.tanh === "undefined") {\r\n    Math.tanh = function(x){\r\n      if
(Math.abs(x) < taylor_n_bound) {\r\n        var result = x;\r\n        if (Math.abs(x) > taylor_2_bound) {\r\n

```

```

        result -= (x * x * x) / 3;\r\n    }\r\n        return result;\r\n    }\r\n    else {\r\n
var a = Math.exp(+x), b = Math.exp(-x);\r\n        return a === Infinity ? 1 : b === Infinity ? -1 : (a - b) / (a +
b);\r\n    }\r\n    };\r\n    }\r\n\r\n    // Inverse hyperbolic function implementations derived from boost special
math functions,\r\n    // Copyright Eric Ford & Hubert Holin 2001.\r\n\r\n    if (typeof Math.asinh === "undefined")
{\r\n        var asinh = function(x) {\r\n            if (x >= +taylor_n_bound)\r\n                {\r\n                    if (x >
upper_taylor_n_bound)\r\n                        {\r\n                            if (x > upper_taylor_2_bound)\r\n                                {\r\n
// approximation by laurent series in 1/x at 0+ order from -1 to 0\r\n                                    return Math.log(x) +
Math.LN2;\r\n                                }\r\n                            else\r\n                                    {\r\n
// approximation by laurent
series in 1/x at 0+ order from -1 to 1\r\n                                        return Math.log(x * 2 + (1 / (x * 2)));\r\n                                    }\r\n
                                }\r\n                            else\r\n                                    {\r\n
return Math.log(x * 2 + (1 / (x * 2)));\r\n                                }\r\n
                                }\r\n                            else if (x <= -taylor_n_bound)\r\n                                {\r\n
return Math.log(x + Math.sqrt(x * x + 1));\r\n                                }\r\n
                                }\r\n                            else if (x <= -taylor_n_bound)\r\n                                {\r\n
return -asinh(-x);\r\n                                }\r\n
                                }\r\n                            // approximation by taylor series in x at 0 up to order 2\r\n                                var result = x;\r\n
                                if (Math.abs(x) >= taylor_2_bound)\r\n                                    {\r\n
var x3 = x * x * x;\r\n                                    //
approximation by taylor series in x at 0 up to order 4\r\n                                        result -= x3 / 6;\r\n                                    }\r\n
return result;\r\n                                }\r\n                                };\r\n                                Math.asinh = asinh;\r\n                                }\r\n                                if (typeof Math.acosh ===
"undefined") {\r\n                                    Math.acosh = function(x) {\r\n                                        if (x < 1)\r\n                                            {\r\n
return NaN;\r\n                                            }\r\n
                                        }\r\n                                        else if (x - 1 >= taylor_n_bound)\r\n                                            {\r\n
if (x > upper_taylor_2_bound)\r\n                                                {\r\n
// approximation by laurent series in 1/x at 0+ order from -1 to 0\r\n                                                    return Math.log(x)
+ Math.LN2;\r\n                                                }\r\n
                                            }\r\n                                        else\r\n                                            {\r\n
return Math.log(x + Math.sqrt(x * x -
1));\r\n                                            }\r\n
                                        }\r\n                                        }\r\n                                        else\r\n                                            {\r\n
var y = Math.sqrt(x - 1);\r\n                                            //
approximation by taylor series in y at 0 up to order 2\r\n                                                var result = y;\r\n                                                if (y >=
taylor_2_bound)\r\n                                                    {\r\n
var y3 = y * y * y;\r\n                                                    // approximation by taylor series in
y at 0 up to order 4\r\n                                                        result -= y3 / 12;\r\n                                                    }\r\n
return Math.sqrt(2) * result;\r\n                                                    }\r\n
                                                    }\r\n                                }\r\n                                if (typeof Math.atanh === "undefined") {\r\n                                    Math.atanh = function(x) {\r\n
if (Math.abs(x) < taylor_n_bound) {\r\n                                        var result = x;\r\n                                        if (Math.abs(x) > taylor_2_bound)
{\r\n                                            result += (x * x * x) / 3;\r\n                                        }\r\n                                        return result;\r\n                                    }\r\n
return
Math.log((1 + x) / (1 - x)) / 2;\r\n                                }\r\n                                }\r\n                                if (typeof Math.log1p === "undefined") {\r\n                                    Math.log1p
= function(x) {\r\n                                        if (Math.abs(x) < taylor_n_bound) {\r\n                                            var x2 = x * x;\r\n                                            var x3 = x2
* x;\r\n                                            var x4 = x3 * x;\r\n                                            // approximation by taylor series in x at 0 up to order 4\r\n
return (-x4 / 4 + x3 / 3 - x2 / 2 + x);\r\n                                        }\r\n                                        return Math.log(x + 1);\r\n                                    }\r\n
                                }\r\n                                }\r\n                                if (typeof Math.expm1 === "undefined") {\r\n                                    Math.expm1 = function(x) {\r\n                                        if (Math.abs(x) < taylor_n_bound)
{\r\n                                            var x2 = x * x;\r\n                                            var x3 = x2 * x;\r\n                                            var x4 = x3 * x;\r\n                                            //
approximation by taylor series in x at 0 up to order 4\r\n                                                return (x4 / 24 + x3 / 6 + x2 / 2 + x);\r\n                                        }\r\n
return Math.exp(x) - 1;\r\n                                    }\r\n                                }\r\n                                }\r\n                                }\r\n                                if (typeof Math.hypot === "undefined") {\r\n                                    Math.hypot = function() {\r\n                                        var y = 0;\r\n                                        var length = arguments.length;\r\n                                        for (var i = 0; i <
length; i++) {\r\n                                            if (arguments[i] === Infinity || arguments[i] === -Infinity) {\r\n                                                return Infinity;\r\n
                                            }\r\n                                            y += arguments[i] * arguments[i];\r\n                                        }\r\n                                        return Math.sqrt(y);\r\n                                    }\r\n
                                }\r\n                                }\r\n                                if (typeof Math.log10 === "undefined") {\r\n                                    Math.log10 = function(x) {\r\n                                        return Math.log(x) * Math.LOG10E;\r\n
                                    }\r\n                                }\r\n                                }\r\n                                if (typeof Math.log2 === "undefined") {\r\n                                    Math.log2 = function(x) {\r\n                                        return Math.log(x) *
Math.LOG2E;\r\n                                    }\r\n                                }\r\n                                }\r\n                                if (typeof Math.clz32 === "undefined") {\r\n                                    Math.clz32 = (function(log, LN2)
{\r\n                                        return function(x) {\r\n                                            var asUInt = x >>> 0;\r\n                                            if (asUInt === 0) {\r\n                                                return
32;\r\n                                            }\r\n                                            return 31 - (log(asUInt) / LN2 | 0) | 0; // the "|" 0" acts like math.floor\r\n                                        }\r\n
                                    })(Math.log, Math.LN2);\r\n                                }\r\n                                }\r\n                                // For HtmlUnit and PhantomJs\r\n                                if (typeof ArrayBuffer.isView ===
"undefined") {\r\n                                    ArrayBuffer.isView = function(a) {\r\n                                        return a != null && a.__proto__ != null &&
a.__proto__.__proto__ === Int8Array.prototype.__proto__;\r\n                                    }\r\n                                }\r\n                                }\r\n                                if (typeof Array.prototype.fill ===
"undefined") {\r\n                                    // Polyfill from https://developer.mozilla.org/en-
US/docs/Web/JavaScript/Reference/Global_Objects/Array/fill#Polyfill\r\n                                    Array.prototype.fill = function() {\r\n
// Steps 1-2.\r\n                                    if (this == null) {\r\n                                        throw new TypeError('this is null or not defined');\r\n                                    }\r\n
                                }\r\n

```

```

var O = Object(this);
// Steps 3-5.
var len = O.length >>> 0;
// Steps 6-7.
var start = arguments[1];
var relativeStart = start >> 0;
// Step 8.
var k = relativeStart < 0 ?
Math.max(len + relativeStart, 0) :
Math.min(relativeStart, len);
// Steps 9-10.
var end = arguments[2];
var relativeEnd = end === undefined ?
len : end >> 0;
// Step 11.
var final = relativeEnd < 0 ?
Math.max(len + relativeEnd, 0) :
Math.min(relativeEnd, len);
// Step 12.
while (k < final) {
O[k] = value;
k++;
}
// Step 13.
return O;
}
}(function() {
function normalizeOffset(offset, length) {
if (offset < 0) return
Math.max(0, offset + length);
return Math.min(offset, length);
}
function typedArraySlice(begin, end) {
if (typeof end === "undefined") {
end = this.length;
}
begin =
normalizeOffset(begin || 0, this.length);
end = Math.max(begin, normalizeOffset(end, this.length));
return new this.constructor(this.subarray(begin, end));
}
var arrays = [Int8Array, Int16Array,
Uint16Array, Int32Array, Float32Array, Float64Array];
for (var i = 0; i < arrays.length; ++i) {
var
TypedArray = arrays[i];
if (typeof TypedArray.prototype.fill === "undefined") {
TypedArray.prototype.fill = Array.prototype.fill;
}
if (typeof TypedArray.prototype.slice ===
"undefined") {
Object.defineProperty(TypedArray.prototype, 'slice', {
value:
typedArraySlice
});
}
// Patch apply to work with TypedArrays if needed.
try {
(function() {}).apply(null, new Int32Array(0))
} catch (e) {
var apply =
Function.prototype.apply;
Object.defineProperty(Function.prototype, 'apply', {
value:
function(self, array) {
return apply.call(this, self, [].slice.call(array));
}
});
}
// Patch map to work with TypedArrays if needed.
for (var i = 0; i < arrays.length; ++i) {
var TypedArray = arrays[i];
if (typeof TypedArray.prototype.map === "undefined") {
Object.defineProperty(TypedArray.prototype, 'map', {
value: function(callback, self) {
return [].slice.call(this).map(callback, self);
}
});
}
// Patch sort to
work with TypedArrays if needed.
// TODO: consider to remove following function and replace it with
`Kotlin.doubleCompareTo` (see misc.js)
var totalOrderComparator = function (a, b) {
if (a < b) return -
1;
if (a > b) return 1;
if (a === b) {
if (a !== 0) return 0;
var ia = 1 / a;
return ia === 1 / b ? 0 : (ia < 0 ? -1 : 1);
}
return a !== a ? (b !== b ? 0 : 1) : -1
};
for (var i = 0; i < arrays.length; ++i) {
var TypedArray = arrays[i];
if (typeof
TypedArray.prototype.sort === "undefined") {
Object.defineProperty(TypedArray.prototype, 'sort', {
value: function(compareFunction) {
return Array.prototype.sort.call(this, compareFunction ||
totalOrderComparator);
}
});
}
}
}
}

/*
 * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.
 *
 */
Kotlin.Kind = {
CLASS:
"class",
INTERFACE: "interface",
OBJECT: "object"
};
Kotlin.callGetter = function
(thisObject, klass, propertyName) {
var propertyDescriptor = Object.getOwnPropertyDescriptor(klass,
propertyName);
if (propertyDescriptor != null && propertyDescriptor.get != null) {
return
propertyDescriptor.get.call(thisObject);
}
propertyDescriptor =
Object.getOwnPropertyDescriptor(thisObject, propertyName);
if (propertyDescriptor != null && "value" in
propertyDescriptor) {
return thisObject[propertyName];
}
return Kotlin.callGetter(thisObject,
Object.getPrototypeOf(klass), propertyName);
}
Kotlin.callSetter = function (thisObject, klass,
propertyName, value) {
var propertyDescriptor = Object.getOwnPropertyDescriptor(klass, propertyName);
if (propertyDescriptor != null && propertyDescriptor.set != null) {
propertyDescriptor.set.call(thisObject,
value);
return;
}
propertyDescriptor = Object.getOwnPropertyDescriptor(thisObject,
propertyName);
if (propertyDescriptor != null && "value" in propertyDescriptor) {
thisObject[propertyName] = value;
return
}
Kotlin.callSetter(thisObject,
Object.getPrototypeOf(klass), propertyName, value);
}
function isInheritanceFromInterface(ctor, iface)
{
if (ctor === iface) return true;
var metadata = ctor.$metadata$;
if (metadata != null) {
var interfaces = metadata.interfaces;
for (var i = 0; i < interfaces.length; i++) {
if

```

```

(isInheritanceFromInterface(interfaces[i], iface)) {\r\n          return true;\r\n        }\r\n    }\r\n\r\n    var
superPrototype = ctor.prototype != null ? Object.getPrototypeOf(ctor.prototype) : null;\r\n    var superConstructor =
superPrototype != null ? superPrototype.constructor : null;\r\n    return superConstructor != null &&
isInheritanceFromInterface(superConstructor, iface);\r\n}\r\n\r\n\r\n**\r\n * @param {*} object\r\n * @param
{Function|Object} klass\r\n * @returns {Boolean}\r\n * \r\nKotlin.isType = function (object, klass) {\r\n    if (klass
=== Object) {\r\n        switch (typeof object) {\r\n            case "string":\r\n                case "number":\r\n                case
"boolean":\r\n                case "function":\r\n                    return true;\r\n                default:\r\n                    return object
instanceof Object;\r\n            }\r\n        }\r\n\r\n        if (object == null || klass == null || (typeof object !== 'object' && typeof
object !== 'function')) {\r\n            return false;\r\n        }\r\n\r\n        if (typeof klass === "function" && object instanceof
klass) {\r\n            return true;\r\n        }\r\n\r\n        var proto = Object.getPrototypeOf(klass);\r\n        var constructor = proto
!= null ? proto.constructor : null;\r\n        if (constructor != null && {"$metadata$" in constructor}) {\r\n            var
metadata = constructor.$metadata$;\r\n            if (metadata.kind === Kotlin.Kind.OBJECT) {\r\n                return object
=== klass;\r\n            }\r\n        }\r\n\r\n        var klassMetadata = klass.$metadata$;\r\n\r\n        // In WebKit (JavaScriptCore)
for some interfaces from DOM typeof returns "object", nevertheless they can be used in RHS of instanceof\r\n        if
(klassMetadata == null) {\r\n            return object instanceof klass;\r\n        }\r\n\r\n        if (klassMetadata.kind ===
Kotlin.Kind.INTERFACE && object.constructor != null) {\r\n            return
isInheritanceFromInterface(object.constructor, klass);\r\n        }\r\n\r\n        return false;\r\n    };\r\n\r\n\r\nKotlin.isNumber =
function (a) {\r\n    return typeof a == "number" || a instanceof Kotlin.Long;\r\n};\r\n\r\n\r\nKotlin.isChar = function
(value) {\r\n    return value instanceof Kotlin.BoxedChar;\r\n};\r\n\r\n\r\nKotlin.isComparable = function (value) {\r\n
var type = typeof value;\r\n\r\n    return type === "string" ||\r\n        type === "boolean" ||\r\n
Kotlin.isNumber(value) ||\r\n        Kotlin.isType(value,
Kotlin.kotlin.Comparable);\r\n};\r\n\r\n\r\n\r\nKotlin.isCharSequence = function (value) {\r\n    return typeof value ===
"string" || Kotlin.isType(value, Kotlin.kotlin.CharSequence);\r\n};\r\n";\r\n, null, /*\r\n * Copyright 2010-2018 JetBrains
s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0
license that can be found in the license/LICENSE.txt file.\r\n * \r\n\r\n\r\npackage kotlin\r\n\r\npublic class Enum<T> :
Enum<T>> : Comparable<Enum<T>> {\r\n    @JsName("name$") private var _name: String = ""\r\n
@JsName("ordinal$") private var _ordinal: Int = 0\r\n\r\n    val name: String\r\n        get() = _name\r\n\r\n    val
ordinal: Int\r\n        get() = _ordinal\r\n\r\n    override fun compareTo(other: Enum<T>) =
ordinal.compareTo(other.ordinal)\r\n\r\n    override fun equals(other: Any?) = this === other\r\n\r\n    override fun
hashCode(): Int = js("Kotlin.identityHashCode")(this)\r\n\r\n    override fun toString() = name\r\n\r\n    companion
object\r\n    ", null, /*\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n *
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*\r\n\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("ArraysKt")\r\n\r\npackage
kotlin.collections\r\n\r\n\r\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\r\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\r\n\r\n\r\nimport kotlin.random.*\r\nimport
kotlin.ranges.contains\r\nimport kotlin.ranges.reversed\r\n\r\n\r\n**\r\n * Returns 1st *element* from the collection.\r\n
*\r\n * \r\n@kotlin.internal.InlineOnly\r\npublic inline operator fun <T> Array<out T>.component1(): T {\r\n    return
get(0)\r\n}\r\n\r\n**\r\n * Returns 1st *element* from the collection.\r\n * \r\n@kotlin.internal.InlineOnly\r\npublic inline
operator fun ByteArray.component1(): Byte {\r\n    return get(0)\r\n}\r\n\r\n**\r\n * Returns 1st *element* from the
collection.\r\n * \r\n@kotlin.internal.InlineOnly\r\npublic inline operator fun ShortArray.component1(): Short {\r\n
return get(0)\r\n}\r\n\r\n**\r\n * Returns 1st *element* from the collection.\r\n * \r\n@kotlin.internal.InlineOnly\r\npublic
inline operator fun IntArray.component1(): Int {\r\n    return get(0)\r\n}\r\n\r\n**\r\n * Returns 1st *element* from the
collection.\r\n * \r\n@kotlin.internal.InlineOnly\r\npublic inline operator fun LongArray.component1(): Long {\r\n
return get(0)\r\n}\r\n\r\n**\r\n * Returns 1st *element* from the collection.\r\n * \r\n@kotlin.internal.InlineOnly\r\npublic
inline operator fun FloatArray.component1(): Float {\r\n    return get(0)\r\n}\r\n\r\n**\r\n * Returns 1st *element* from the
collection.\r\n * \r\n@kotlin.internal.InlineOnly\r\npublic inline operator fun DoubleArray.component1(): Double {\r\n
return get(0)\r\n}\r\n\r\n**\r\n * Returns 1st *element* from the collection.\r\n * \r\n@kotlin.internal.InlineOnly\r\npublic
inline operator fun BooleanArray.component1(): Boolean {\r\n    return get(0)\r\n}\r\n\r\n**\r\n * Returns 1st *element*

```



```

return get(4)\n\n/**\n * Returns 5th *element* from the collection.\n *\n@kotlin.internal.InlineOnly\npublic
inline operator fun IntArray.component5(): Int {\n    return get(4)\n}\n\n/**\n * Returns 5th *element* from the
collection.\n *\n@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component5(): Long {\n
return get(4)\n}\n\n/**\n * Returns 5th *element* from the collection.\n *\n@kotlin.internal.InlineOnly\npublic
inline operator fun FloatArray.component5(): Float {\n    return get(4)\n}\n\n/**\n * Returns 5th *element* from the
collection.\n *\n@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component5(): Double {\n
return get(4)\n}\n\n/**\n * Returns 5th *element* from the collection.\n *\n@kotlin.internal.InlineOnly\npublic
inline operator fun BooleanArray.component5(): Boolean {\n    return get(4)\n}\n\n/**\n * Returns 5th *element*
from the collection.\n *\n@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component5(): Char
{\n    return get(4)\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n *\npublic operator fun
<@kotlin.internal.OnlyInputTypes T> Array<out T>.contains(element: T): Boolean {\n    return indexOf(element)
>= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n *\npublic operator fun
ByteArray.contains(element: Byte): Boolean {\n    return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if
[element] is found in the array.\n *\npublic operator fun ShortArray.contains(element: Short): Boolean {\n
return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n *\npublic operator fun
IntArray.contains(element: Int): Boolean {\n    return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if
[element] is found in the array.\n *\npublic operator fun LongArray.contains(element: Long): Boolean {\n    return
indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n *\npublic operator fun
FloatArray.contains(element: Float): Boolean {\n    return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if
[element] is found in the array.\n *\npublic operator fun DoubleArray.contains(element: Double): Boolean {\n
return indexOf(element) >= 0\n}\n\n/**\n * Returns `true` if [element] is found in the array.\n *\npublic operator
fun BooleanArray.contains(element: Boolean): Boolean {\n    return indexOf(element) >= 0\n}\n\n/**\n * Returns
`true` if [element] is found in the array.\n *\npublic operator fun CharArray.contains(element: Char): Boolean {\n
return indexOf(element) >= 0\n}\n\n/**\n * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n *\n\n * @sample
samples.collections.Collections.Elements.elementAt\n *\npublic expect fun <T> Array<out T>.elementAt(index:
Int): T\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index]
is out of bounds of this array.\n *\n\n * @sample samples.collections.Collections.Elements.elementAt\n *\npublic
expect fun ByteArray.elementAt(index: Int): Byte\n\n/**\n * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n *\n\n * @sample
samples.collections.Collections.Elements.elementAt\n *\npublic expect fun ShortArray.elementAt(index: Int):
Short\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index]
is out of bounds of this array.\n *\n\n * @sample samples.collections.Collections.Elements.elementAt\n *\npublic
expect fun IntArray.elementAt(index: Int): Int\n\n/**\n * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n *\n\n * @sample
samples.collections.Collections.Elements.elementAt\n *\npublic expect fun LongArray.elementAt(index: Int):
Long\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index]
is out of bounds of this array.\n *\n\n * @sample samples.collections.Collections.Elements.elementAt\n *\npublic
expect fun FloatArray.elementAt(index: Int): Float\n\n/**\n * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n *\n\n * @sample
samples.collections.Collections.Elements.elementAt\n *\npublic expect fun DoubleArray.elementAt(index: Int):
Double\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the
[index] is out of bounds of this array.\n *\n\n * @sample samples.collections.Collections.Elements.elementAt\n
*\npublic expect fun BooleanArray.elementAt(index: Int): Boolean\n\n/**\n * Returns an element at the given
[index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n *\n\n * @sample
samples.collections.Collections.Elements.elementAt\n *\npublic expect fun CharArray.elementAt(index: Int):
Char\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
[index] is out of bounds of this array.\n *\n\n * @sample

```



```

samples.collections.Collections.Elements.elementAtOrElse\n *\n@kotlin.internal.InlineOnly\npublic inline fun
<T> Array<out T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n *\n@kotlin.internal.InlineOnly\npublic inline fun
ByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Byte): Byte {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n *\n@kotlin.internal.InlineOnly\npublic inline fun
ShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Short): Short {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n *\n@kotlin.internal.InlineOnly\npublic inline fun
IntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Int): Int {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n *\n@kotlin.internal.InlineOnly\npublic inline fun
LongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Long): Long {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n *\n@kotlin.internal.InlineOnly\npublic inline fun
FloatArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Float): Float {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n *\n@kotlin.internal.InlineOnly\npublic inline fun
DoubleArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Double): Double {\n  return if (index >= 0 &&
index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the
result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n *\n@kotlin.internal.InlineOnly\npublic inline fun
BooleanArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {\n  return if (index >= 0 &&
index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the
result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n *\n@kotlin.internal.InlineOnly\npublic inline fun
CharArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Char): Char {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the
[index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n *\n@kotlin.internal.InlineOnly\npublic inline fun
<T> Array<out T>.elementAtOrNull(index: Int): T? {\n  return this.getOrNull(index)\n}\n\n/**\n * Returns an
element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n *\n@kotlin.internal.InlineOnly\npublic inline fun
ByteArray.elementAtOrNull(index: Int): Byte? {\n  return this.getOrNull(index)\n}\n\n/**\n * Returns an element
at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n *\n@kotlin.internal.InlineOnly\npublic inline fun
ShortArray.elementAtOrNull(index: Int): Short? {\n  return this.getOrNull(index)\n}\n\n/**\n * Returns an element
at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n *\n@kotlin.internal.InlineOnly\npublic inline fun
IntArray.elementAtOrNull(index: Int): Int? {\n  return this.getOrNull(index)\n}\n\n/**\n * Returns an element at
the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample

```

```

samples.collections.Collections.Elements.elementAtOrNull\n *^n@kotlin.internal.InlineOnly\npublic inline fun
LongArray.elementAtOrNull(index: Int): Long? {\n  return this.getOrNull(index)\n}\n\n/**\n * Returns an element
at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n *^n@kotlin.internal.InlineOnly\npublic inline fun
FloatArray.elementAtOrNull(index: Int): Float? {\n  return this.getOrNull(index)\n}\n\n/**\n * Returns an element
at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n *^n@kotlin.internal.InlineOnly\npublic inline fun
DoubleArray.elementAtOrNull(index: Int): Double? {\n  return this.getOrNull(index)\n}\n\n/**\n * Returns an
element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n *^n@kotlin.internal.InlineOnly\npublic inline fun
BooleanArray.elementAtOrNull(index: Int): Boolean? {\n  return this.getOrNull(index)\n}\n\n/**\n * Returns an
element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n *^n@kotlin.internal.InlineOnly\npublic inline fun
CharArray.elementAtOrNull(index: Int): Char? {\n  return this.getOrNull(index)\n}\n\n/**\n * Returns the first
element matching the given [predicate], or `null` if no such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.find(predicate: (T) -> Boolean): T? {\n
return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no
such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.find(predicate: (Byte) ->
Boolean): Byte? {\n  return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given
[predicate], or `null` if no such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun
ShortArray.find(predicate: (Short) -> Boolean): Short? {\n  return firstOrNull(predicate)\n}\n\n/**\n * Returns the
first element matching the given [predicate], or `null` if no such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun IntArray.find(predicate: (Int) -> Boolean): Int? {\n  return
firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no such
element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun LongArray.find(predicate: (Long) ->
Boolean): Long? {\n  return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given
[predicate], or `null` if no such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun
FloatArray.find(predicate: (Float) -> Boolean): Float? {\n  return firstOrNull(predicate)\n}\n\n/**\n * Returns the
first element matching the given [predicate], or `null` if no such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.find(predicate: (Double) -> Boolean): Double? {\n
return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if no
such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.find(predicate:
(Boolean) -> Boolean): Boolean? {\n  return firstOrNull(predicate)\n}\n\n/**\n * Returns the first element
matching the given [predicate], or `null` if no such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic
inline fun CharArray.find(predicate: (Char) -> Boolean): Char? {\n  return firstOrNull(predicate)\n}\n\n/**\n *
Returns the last element matching the given [predicate], or `null` if no such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.findLast(predicate: (T) -> Boolean): T? {\n
return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no
such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.findLast(predicate: (Byte) -
> Boolean): Byte? {\n  return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given
[predicate], or `null` if no such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun
ShortArray.findLast(predicate: (Short) -> Boolean): Short? {\n  return lastOrNull(predicate)\n}\n\n/**\n * Returns
the last element matching the given [predicate], or `null` if no such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun IntArray.findLast(predicate: (Int) -> Boolean): Int? {\n  return
lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such
element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun LongArray.findLast(predicate: (Long) ->
Boolean): Long? {\n  return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given
[predicate], or `null` if no such element was found.\n
*^n@kotlin.internal.InlineOnly\npublic inline fun

```

```

FloatArray.findLast(predicate: (Float) -> Boolean): Float? {\n  return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n *\n * @kotlin.internal.InlineOnly\npublic inline fun DoubleArray.findLast(predicate: (Double) -> Boolean): Double? {\n  return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n *\n * @kotlin.internal.InlineOnly\npublic inline fun BooleanArray.findLast(predicate: (Boolean) -> Boolean): Boolean? {\n  return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n *\n * @kotlin.internal.InlineOnly\npublic inline fun CharArray.findLast(predicate: (Char) -> Boolean): Char? {\n  return lastOrNull(predicate)\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n *\npublic fun <T> Array<out T>.first(): T {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n *\npublic fun ByteArray.first(): Byte {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n *\npublic fun ShortArray.first(): Short {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n *\npublic fun IntArray.first(): Int {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n *\npublic fun LongArray.first(): Long {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n *\npublic fun FloatArray.first(): Float {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n *\npublic fun DoubleArray.first(): Double {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n *\npublic fun BooleanArray.first(): Boolean {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return this[0]\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n *\npublic fun CharArray.first(): Char {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return this[0]\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n *\npublic inline fun <T> Array<out T>.first(predicate: (T) -> Boolean): T {\n  for (element in this) if (predicate(element)) return element\n  throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n *\npublic inline fun ByteArray.first(predicate: (Byte) -> Boolean): Byte {\n  for (element in this) if (predicate(element)) return element\n  throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n *\npublic inline fun ShortArray.first(predicate: (Short) -> Boolean): Short {\n  for (element in this) if (predicate(element)) return element\n  throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n *\npublic inline fun IntArray.first(predicate: (Int) -> Boolean): Int {\n  for (element in this) if (predicate(element)) return element\n  throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n *\npublic inline fun LongArray.first(predicate: (Long) -> Boolean): Long {\n  for (element in this) if (predicate(element)) return element\n  throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n *\npublic inline fun FloatArray.first(predicate: (Float) -> Boolean): Float {\n  for (element in this) if (predicate(element)) return element\n  throw NoSuchElementException("Array contains no element matching the

```

predicate.)\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun DoubleArray.first(predicate: (Double) -> Boolean): Double {\n for (element in this) if (predicate(element)) return element\n throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun BooleanArray.first(predicate: (Boolean) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return element\n throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun CharArray.first(predicate: (Char) -> Boolean): Char {\n for (element in this) if (predicate(element)) return element\n throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun <T> Array<out T>.firstOrNull(): T? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun ByteArray.firstOrNull(): Byte? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun ShortArray.firstOrNull(): Short? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun IntArray.firstOrNull(): Int? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun LongArray.firstOrNull(): Long? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun FloatArray.firstOrNull(): Float? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun DoubleArray.firstOrNull(): Double? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n */\npublic fun BooleanArray.firstOrNull(): Boolean? {\n return if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun <T> Array<out T>.firstOrNull(predicate: (T) -> Boolean): T? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun ByteArray.firstOrNull(predicate: (Byte) -> Boolean): Byte? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun ShortArray.firstOrNull(predicate: (Short) -> Boolean): Short? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun IntArray.firstOrNull(predicate: (Int) -> Boolean): Int? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun LongArray.firstOrNull(predicate: (Long) -> Boolean): Long? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun FloatArray.firstOrNull(predicate: (Float) -> Boolean): Float? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun DoubleArray.firstOrNull(predicate: (Double) -> Boolean): Double? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun BooleanArray.firstOrNull(predicate: (Boolean) -> Boolean): Boolean? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun CharArray.firstOrNull(predicate: (Char) -> Boolean): Char? {\n for (element in this) if (predicate(element)) return element\n return null\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.getOrElse(index: Int,

```

defaultValue: (Int) -> T): T {\n  return if (index >= 0 && index <= lastIndex) get(index) else
defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue]
function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun
ByteArray.getOrNull(index: Int, defaultValue: (Int) -> Byte): Byte {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n
*/\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.getOrNull(index: Int, defaultValue: (Int) -> Short):
Short {\n  return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns
an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of
this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.getOrNull(index: Int, defaultValue: (Int) ->
Int): Int {\n  return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n *
Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of
bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.getOrNull(index: Int,
defaultValue: (Int) -> Long): Long {\n  return if (index >= 0 && index <= lastIndex) get(index) else
defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue]
function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun
FloatArray.getOrNull(index: Int, defaultValue: (Int) -> Float): Float {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n
*/\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.getOrNull(index: Int, defaultValue: (Int) ->
Double): Double {\n  return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n
* Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of
bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.getOrNull(index: Int,
defaultValue: (Int) -> Boolean): Boolean {\n  return if (index >= 0 && index <= lastIndex) get(index) else
defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue]
function if the [index] is out of bounds of this array.\n */\n@kotlin.internal.InlineOnly\npublic inline fun
CharArray.getOrNull(index: Int, defaultValue: (Int) -> Char): Char {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the
[index] is out of bounds of this array.\n */\npublic fun <T> Array<out T>.getOrNull(index: Int): T? {\n  return if
(index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or
`null` if the [index] is out of bounds of this array.\n */\npublic fun ByteArray.getOrNull(index: Int): Byte? {\n
return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given
[index] or `null` if the [index] is out of bounds of this array.\n */\npublic fun ShortArray.getOrNull(index: Int):
Short? {\n  return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at
the given [index] or `null` if the [index] is out of bounds of this array.\n */\npublic fun IntArray.getOrNull(index:
Int): Int? {\n  return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element
at the given [index] or `null` if the [index] is out of bounds of this array.\n */\npublic fun LongArray.getOrNull(index:
Int): Long? {\n  return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element
at the given [index] or `null` if the [index] is out of bounds of this array.\n */\npublic fun
FloatArray.getOrNull(index: Int): Float? {\n  return if (index >= 0 && index <= lastIndex) get(index) else
null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n
*/\npublic fun DoubleArray.getOrNull(index: Int): Double? {\n  return if (index >= 0 && index <= lastIndex)
get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of
this array.\n */\npublic fun BooleanArray.getOrNull(index: Int): Boolean? {\n  return if (index >= 0 && index <=
lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of
bounds of this array.\n */\npublic fun CharArray.getOrNull(index: Int): Char? {\n  return if (index >= 0 && index
<= lastIndex) get(index) else null\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain
element.\n */\npublic fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.indexOf(element: T): Int {\n  if

```

```

(element == null) {\n    for (index in indices) {\n        if (this[index] == null) {\n            return index\n        }\n    } else {\n        for (index in indices) {\n            if (element == this[index]) {\n                return index\n            }\n        }\n    } return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun ByteArray.indexOf(element: Byte): Int {\n    for (index in indices) {\n        if (element == this[index]) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun ShortArray.indexOf(element: Short): Int {\n    for (index in indices) {\n        if (element == this[index]) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun IntArray.indexOf(element: Int): Int {\n    for (index in indices) {\n        if (element == this[index]) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun LongArray.indexOf(element: Long): Int {\n    for (index in indices) {\n        if (element == this[index]) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun FloatArray.indexOf(element: Float): Int {\n    for (index in indices) {\n        if (element == this[index]) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun DoubleArray.indexOf(element: Double): Int {\n    for (index in indices) {\n        if (element == this[index]) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun BooleanArray.indexOf(element: Boolean): Int {\n    for (index in indices) {\n        if (element == this[index]) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\npublic fun CharArray.indexOf(element: Char): Int {\n    for (index in indices) {\n        if (element == this[index]) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun <T> Array<out T>.indexOfFirst(predicate: (T) -> Boolean): Int {\n    for (index in indices) {\n        if (predicate(this[index])) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun ByteArray.indexOfFirst(predicate: (Byte) -> Boolean): Int {\n    for (index in indices) {\n        if (predicate(this[index])) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun ShortArray.indexOfFirst(predicate: (Short) -> Boolean): Int {\n    for (index in indices) {\n        if (predicate(this[index])) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun IntArray.indexOfFirst(predicate: (Int) -> Boolean): Int {\n    for (index in indices) {\n        if (predicate(this[index])) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun LongArray.indexOfFirst(predicate: (Long) -> Boolean): Int {\n    for (index in indices) {\n        if (predicate(this[index])) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun FloatArray.indexOfFirst(predicate: (Float) -> Boolean): Int {\n    for (index in indices) {\n        if (predicate(this[index])) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun DoubleArray.indexOfFirst(predicate: (Double) -> Boolean): Int {\n    for (index in indices) {\n        if (predicate(this[index])) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun BooleanArray.indexOfFirst(predicate: (Boolean) -> Boolean): Int {\n    for (index in indices) {\n        if (predicate(this[index])) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun CharArray.indexOfFirst(predicate: (Char) -> Boolean): Int {\n    for (index in indices) {\n        if (predicate(this[index])) {\n            return index\n        }\n    } return -1\n}\n\n/**\n * Returns index of the last

```

```

element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun <T>
Array<out T>.indexOfLast(predicate: (T) -> Boolean): Int {\n    for (index in indices.reversed()) {\n        if
(predicate(this[index])) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns index of the last
element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun
ByteArray.indexOfLast(predicate: (Byte) -> Boolean): Int {\n    for (index in indices.reversed()) {\n        if
(predicate(this[index])) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns index of the last
element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun
ShortArray.indexOfLast(predicate: (Short) -> Boolean): Int {\n    for (index in indices.reversed()) {\n        if
(predicate(this[index])) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns index of the last
element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun
IntArray.indexOfLast(predicate: (Int) -> Boolean): Int {\n    for (index in indices.reversed()) {\n        if
(predicate(this[index])) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns index of the last
element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun
LongArray.indexOfLast(predicate: (Long) -> Boolean): Int {\n    for (index in indices.reversed()) {\n        if
(predicate(this[index])) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns index of the last
element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun
FloatArray.indexOfLast(predicate: (Float) -> Boolean): Int {\n    for (index in indices.reversed()) {\n        if
(predicate(this[index])) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns index of the last
element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun
DoubleArray.indexOfLast(predicate: (Double) -> Boolean): Int {\n    for (index in indices.reversed()) {\n        if
(predicate(this[index])) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns index of the last
element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun
BooleanArray.indexOfLast(predicate: (Boolean) -> Boolean): Int {\n    for (index in indices.reversed()) {\n        if
(predicate(this[index])) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns index of the last
element matching the given [predicate], or -1 if the array does not contain such element.\n */\npublic inline fun
CharArray.indexOfLast(predicate: (Char) -> Boolean): Int {\n    for (index in indices.reversed()) {\n        if
(predicate(this[index])) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns the last element.\n *
@throws [NoSuchElementException] if the array is empty.\n */\npublic fun <T> Array<out T>.last(): T {\n    if
(isEmpty())\n        throw NoSuchElementException("Array is empty.")\n    return this[lastIndex]\n}\n\n/**\n *
Returns the last element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun
ByteArray.last(): Byte {\n    if (isEmpty())\n        throw NoSuchElementException("Array is empty.")\n    return
this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * @throws [NoSuchElementException] if the array is
empty.\n */\npublic fun ShortArray.last(): Short {\n    if (isEmpty())\n        throw NoSuchElementException("Array
is empty.")\n    return this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * @throws
[NoSuchElementException] if the array is empty.\n */\npublic fun IntArray.last(): Int {\n    if (isEmpty())\n        throw
NoSuchElementException("Array is empty.")\n    return this[lastIndex]\n}\n\n/**\n * Returns the last
element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun LongArray.last(): Long {\n    if
(isEmpty())\n        throw NoSuchElementException("Array is empty.")\n    return this[lastIndex]\n}\n\n/**\n *
Returns the last element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun
FloatArray.last(): Float {\n    if (isEmpty())\n        throw NoSuchElementException("Array is empty.")\n    return
this[lastIndex]\n}\n\n/**\n * Returns the last element.\n * @throws [NoSuchElementException] if the array is
empty.\n */\npublic fun DoubleArray.last(): Double {\n    if (isEmpty())\n        throw
NoSuchElementException("Array is empty.")\n    return this[lastIndex]\n}\n\n/**\n * Returns the last element.\n *
@throws [NoSuchElementException] if the array is empty.\n */\npublic fun BooleanArray.last(): Boolean {\n    if
(isEmpty())\n        throw NoSuchElementException("Array is empty.")\n    return this[lastIndex]\n}\n\n/**\n *
Returns the last element.\n * @throws [NoSuchElementException] if the array is empty.\n */\npublic fun
CharArray.last(): Char {\n    if (isEmpty())\n        throw NoSuchElementException("Array is empty.")\n    return
this[lastIndex]\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws

```

```

[NoSuchElementException] if no such element is found.\n *\npublic inline fun <T> Array<out T>.last(predicate:
(T) -> Boolean): T {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if
(predicate(element)) return element\n    }\n    throw NoSuchElementException("Array contains no element
matching the predicate.")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws
[NoSuchElementException] if no such element is found.\n *\npublic inline fun ByteArray.last(predicate: (Byte) ->
Boolean): Byte {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if
(predicate(element)) return element\n    }\n    throw NoSuchElementException("Array contains no element
matching the predicate.")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws
[NoSuchElementException] if no such element is found.\n *\npublic inline fun ShortArray.last(predicate: (Short) ->
Boolean): Short {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if
(predicate(element)) return element\n    }\n    throw NoSuchElementException("Array contains no element
matching the predicate.")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws
[NoSuchElementException] if no such element is found.\n *\npublic inline fun IntArray.last(predicate: (Int) ->
Boolean): Int {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if
(predicate(element)) return element\n    }\n    throw NoSuchElementException("Array contains no element
matching the predicate.")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws
[NoSuchElementException] if no such element is found.\n *\npublic inline fun LongArray.last(predicate: (Long) ->
Boolean): Long {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if
(predicate(element)) return element\n    }\n    throw NoSuchElementException("Array contains no element
matching the predicate.")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws
[NoSuchElementException] if no such element is found.\n *\npublic inline fun FloatArray.last(predicate: (Float) ->
Boolean): Float {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if
(predicate(element)) return element\n    }\n    throw NoSuchElementException("Array contains no element
matching the predicate.")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws
[NoSuchElementException] if no such element is found.\n *\npublic inline fun DoubleArray.last(predicate:
(Double) -> Boolean): Double {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if
(predicate(element)) return element\n    }\n    throw NoSuchElementException("Array contains no element
matching the predicate.")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws
[NoSuchElementException] if no such element is found.\n *\npublic inline fun BooleanArray.last(predicate:
(Boolean) -> Boolean): Boolean {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if
(predicate(element)) return element\n    }\n    throw NoSuchElementException("Array contains no element
matching the predicate.")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws
[NoSuchElementException] if no such element is found.\n *\npublic inline fun CharArray.last(predicate: (Char) ->
Boolean): Char {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if
(predicate(element)) return element\n    }\n    throw NoSuchElementException("Array contains no element
matching the predicate.")\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain
element.\n *\npublic fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.lastIndexOf(element: T): Int {\n    if
(element == null) {\n        for (index in indices.reversed()) {\n            if (this[index] == null) {\n                return
index\n            }\n        }\n    } else {\n        for (index in indices.reversed()) {\n            if (element == this[index]) {\n                return
index\n            }\n        }\n    }\n    return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the
array does not contain element.\n *\npublic fun ByteArray.lastIndexOf(element: Byte): Int {\n    for (index in
indices.reversed()) {\n        if (element == this[index]) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n *\npublic fun
ShortArray.lastIndexOf(element: Short): Int {\n    for (index in indices.reversed()) {\n        if (element ==
this[index]) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns last index of [element], or -1 if
the array does not contain element.\n *\npublic fun IntArray.lastIndexOf(element: Int): Int {\n    for (index in
indices.reversed()) {\n        if (element == this[index]) {\n            return index\n        }\n    }\n    return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n *\npublic fun

```



```

LongArray.lastIndexOf(element: Long): Int {\n  for (index in indices.reversed()) {\n    if (element ==
this[index]) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of [element], or -1 if
the array does not contain element.\n */\npublic fun FloatArray.lastIndexOf(element: Float): Int {\n  for (index in
indices.reversed()) {\n    if (element == this[index]) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\npublic fun
DoubleArray.lastIndexOf(element: Double): Int {\n  for (index in indices.reversed()) {\n    if (element ==
this[index]) {\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns last index of [element], or -1 if
the array does not contain element.\n */\npublic fun BooleanArray.lastIndexOf(element: Boolean): Int {\n  for
(index in indices.reversed()) {\n    if (element == this[index]) {\n      return index\n    }\n  }\n  return -
1\n}\n\n/**\n * Returns last index of [element], or -1 if the array does not contain element.\n */\npublic fun
CharArray.lastIndexOf(element: Char): Int {\n  for (index in indices.reversed()) {\n    if (element == this[index])
{\n      return index\n    }\n  }\n  return -1\n}\n\n/**\n * Returns the last element, or `null` if the array is
empty.\n */\npublic fun <T> Array<out T>.lastOrNull(): T? {\n  return if (isEmpty()) null else this[size -
1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n */\npublic fun ByteArray.lastOrNull():
Byte? {\n  return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is
empty.\n */\npublic fun ShortArray.lastOrNull(): Short? {\n  return if (isEmpty()) null else this[size -
1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n */\npublic fun IntArray.lastOrNull(): Int?
{\n  return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is
empty.\n */\npublic fun LongArray.lastOrNull(): Long? {\n  return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n */\npublic fun FloatArray.lastOrNull(): Float? {\n
return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n
*/\npublic fun DoubleArray.lastOrNull(): Double? {\n  return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns
the last element, or `null` if the array is empty.\n */\npublic fun BooleanArray.lastOrNull(): Boolean? {\n
return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n
*/\npublic fun CharArray.lastOrNull(): Char? {\n  return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns
the last element matching the given [predicate], or `null` if no such element was found.\n */\npublic inline fun <T>
Array<out T>.lastOrNull(predicate: (T) -> Boolean): T? {\n  for (index in this.indices.reversed()) {\n    val
element = this[index]\n    if (predicate(element)) return element\n  }\n  return null\n}\n\n/**\n * Returns the last
element matching the given [predicate], or `null` if no such element was found.\n */\npublic inline fun
ByteArray.lastOrNull(predicate: (Byte) -> Boolean): Byte? {\n  for (index in this.indices.reversed()) {\n    val
element = this[index]\n    if (predicate(element)) return element\n  }\n  return null\n}\n\n/**\n * Returns the last
element matching the given [predicate], or `null` if no such element was found.\n */\npublic inline fun
ShortArray.lastOrNull(predicate: (Short) -> Boolean): Short? {\n  for (index in this.indices.reversed()) {\n    val
element = this[index]\n    if (predicate(element)) return element\n  }\n  return null\n}\n\n/**\n * Returns the last
element matching the given [predicate], or `null` if no such element was found.\n */\npublic inline fun
IntArray.lastOrNull(predicate: (Int) -> Boolean): Int? {\n  for (index in this.indices.reversed()) {\n    val element
= this[index]\n    if (predicate(element)) return element\n  }\n  return null\n}\n\n/**\n * Returns the last element
matching the given [predicate], or `null` if no such element was found.\n */\npublic inline fun
LongArray.lastOrNull(predicate: (Long) -> Boolean): Long? {\n  for (index in this.indices.reversed()) {\n    val
element = this[index]\n    if (predicate(element)) return element\n  }\n  return null\n}\n\n/**\n * Returns the last
element matching the given [predicate], or `null` if no such element was found.\n */\npublic inline fun
FloatArray.lastOrNull(predicate: (Float) -> Boolean): Float? {\n  for (index in this.indices.reversed()) {\n    val
element = this[index]\n    if (predicate(element)) return element\n  }\n  return null\n}\n\n/**\n * Returns the last
element matching the given [predicate], or `null` if no such element was found.\n */\npublic inline fun
DoubleArray.lastOrNull(predicate: (Double) -> Boolean): Double? {\n  for (index in this.indices.reversed()) {\n
val element = this[index]\n    if (predicate(element)) return element\n  }\n  return null\n}\n\n/**\n * Returns the
last element matching the given [predicate], or `null` if no such element was found.\n */\npublic inline fun
BooleanArray.lastOrNull(predicate: (Boolean) -> Boolean): Boolean? {\n  for (index in this.indices.reversed()) {\n

```

```

    val element = this[index]\n        if (predicate(element)) return element\n    }\n    return null\n}\n\n/**\n * Returns
the last element matching the given [predicate], or `null` if no such element was found.\n */\npublic inline fun
CharArray.lastOrNull(predicate: (Char) -> Boolean): Char? {\n    for (index in this.indices.reversed()) {\n        val
element = this[index]\n        if (predicate(element)) return element\n    }\n    return null\n}\n\n/**\n * Returns a
random element from this array.\n */\n * @throws NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.random(): T {\n
return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n */\n * @throws
NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.random(): Byte {\n    return random(Random)\n}\n\n/**\n * Returns a random element from
this array.\n */\n * @throws NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.random(): Short {\n    return
random(Random)\n}\n\n/**\n * Returns a random element from this array.\n */\n * @throws
NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.random(): Int {\n    return random(Random)\n}\n\n/**\n * Returns a random element from this
array.\n */\n * @throws NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.random(): Long {\n    return
random(Random)\n}\n\n/**\n * Returns a random element from this array.\n */\n * @throws
NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.random(): Float {\n    return random(Random)\n}\n\n/**\n * Returns a random element from
this array.\n */\n * @throws NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.random(): Double {\n    return
random(Random)\n}\n\n/**\n * Returns a random element from this array.\n */\n * @throws
NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.random(): Boolean {\n    return random(Random)\n}\n\n/**\n * Returns a random element
from this array.\n */\n * @throws NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.random(): Char {\n    return
random(Random)\n}\n\n/**\n * Returns a random element from this array using the specified source of
randomness.\n */\n * @throws NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\npublic fun <T> Array<out T>.random(random: Random): T {\n    if (isEmpty())\n        throw
NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}\n\n/**\n * Returns a random
element from this array using the specified source of randomness.\n */\n * @throws NoSuchElementException if
this array is empty.\n\n*/\n@SinceKotlin("1.3")\npublic fun ByteArray.random(random: Random): Byte {\n    if
(isEmpty())\n        throw NoSuchElementException("Array is empty.")\n    return
get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of
randomness.\n */\n * @throws NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\npublic fun ShortArray.random(random: Random): Short {\n    if (isEmpty())\n        throw
NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}\n\n/**\n * Returns a random
element from this array using the specified source of randomness.\n */\n * @throws NoSuchElementException if
this array is empty.\n\n*/\n@SinceKotlin("1.3")\npublic fun IntArray.random(random: Random): Int {\n    if
(isEmpty())\n        throw NoSuchElementException("Array is empty.")\n    return
get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of
randomness.\n */\n * @throws NoSuchElementException if this array is empty.\n\n*/\n@SinceKotlin("1.3")\npublic fun LongArray.random(random: Random): Long {\n    if (isEmpty())\n        throw
NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}\n\n/**\n * Returns a random
element from this array using the specified source of randomness.\n */\n * @throws NoSuchElementException if
this array is empty.\n\n*/\n@SinceKotlin("1.3")\npublic fun FloatArray.random(random: Random): Float {\n    if
(isEmpty())\n        throw NoSuchElementException("Array is empty.")\n    return
get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of

```

```

randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin("1.3")\npublic
fun DoubleArray.random(random: Random): Double {\n    if (isEmpty())\n        throw
NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}\n\n/**\n * Returns a random
element from this array using the specified source of randomness.\n * \n * @throws NoSuchElementException if
this array is empty.\n */\n@SinceKotlin("1.3")\npublic fun BooleanArray.random(random: Random): Boolean {\n
    if (isEmpty())\n        throw NoSuchElementException("Array is empty.")\n    return
get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of
randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n */\n@SinceKotlin("1.3")\npublic
fun CharArray.random(random: Random): Char {\n    if (isEmpty())\n        throw NoSuchElementException("Array
is empty.")\n    return get(random.nextInt(size))\n}\n\n/**\n * Returns the single element, or throws an exception if
the array is empty or has more than one element.\n */\npublic fun <T> Array<out T>.single(): T {\n    return when
(size) {\n        0 -> throw NoSuchElementException("Array is empty.")\n        1 -> this[0]\n        else -> throw
IllegalArgumentException("Array has more than one element.")\n    }\n}\n\n/**\n * Returns the single element, or
throws an exception if the array is empty or has more than one element.\n */\npublic fun ByteArray.single(): Byte
{\n    return when (size) {\n        0 -> throw NoSuchElementException("Array is empty.")\n        1 -> this[0]\n
else -> throw IllegalArgumentException("Array has more than one element.")\n    }\n}\n\n/**\n * Returns the
single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun
ShortArray.single(): Short {\n    return when (size) {\n        0 -> throw NoSuchElementException("Array is
empty.")\n        1 -> this[0]\n        else -> throw IllegalArgumentException("Array has more than one element.")\n
}\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one
element.\n */\npublic fun IntArray.single(): Int {\n    return when (size) {\n        0 -> throw
NoSuchElementException("Array is empty.")\n        1 -> this[0]\n        else -> throw
IllegalArgumentException("Array has more than one element.")\n    }\n}\n\n/**\n * Returns the single element, or
throws an exception if the array is empty or has more than one element.\n */\npublic fun LongArray.single(): Long
{\n    return when (size) {\n        0 -> throw NoSuchElementException("Array is empty.")\n        1 -> this[0]\n
else -> throw IllegalArgumentException("Array has more than one element.")\n    }\n}\n\n/**\n * Returns the
single element, or throws an exception if the array is empty or has more than one element.\n */\npublic fun
FloatArray.single(): Float {\n    return when (size) {\n        0 -> throw NoSuchElementException("Array is
empty.")\n        1 -> this[0]\n        else -> throw IllegalArgumentException("Array has more than one element.")\n
}\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one
element.\n */\npublic fun DoubleArray.single(): Double {\n    return when (size) {\n        0 -> throw
NoSuchElementException("Array is empty.")\n        1 -> this[0]\n        else -> throw
IllegalArgumentException("Array has more than one element.")\n    }\n}\n\n/**\n * Returns the single element, or
throws an exception if the array is empty or has more than one element.\n */\npublic fun BooleanArray.single():
Boolean {\n    return when (size) {\n        0 -> throw NoSuchElementException("Array is empty.")\n        1 ->
this[0]\n        else -> throw IllegalArgumentException("Array has more than one element.")\n    }\n}\n\n/**\n *
Returns the single element, or throws an exception if the array is empty or has more than one element.\n */\npublic
fun CharArray.single(): Char {\n    return when (size) {\n        0 -> throw NoSuchElementException("Array is
empty.")\n        1 -> this[0]\n        else -> throw IllegalArgumentException("Array has more than one element.")\n
}\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or
more than one matching element.\n */\npublic inline fun <T> Array<out T>.single(predicate: (T) -> Boolean): T {\n
    var single: T? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if
(found) throw IllegalArgumentException("Array contains more than one matching element.")\n            single =
element\n            found = true\n        }\n    }\n    if (!found) throw NoSuchElementException("Array contains no
element matching the predicate.")\n    @Suppress("UNCHECKED_CAST")\n    return single as T\n}\n\n/**\n * Returns
the single element matching the given [predicate], or throws exception if there is no or more than one
matching element.\n */\npublic inline fun ByteArray.single(predicate: (Byte) -> Boolean): Byte {\n    var single:
Byte? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if
(found) throw

```

```

IllegalArgumentException("Array contains more than one matching element.")\n        single = element\n
found = true\n    }\n    }\n    if (!found) throw NoSuchElementException("Array contains no element matching
the predicate.")\n    @Suppress("UNCHECKED_CAST")\n    return single as Byte\n}\n\n/**\n * Returns the
single element matching the given [predicate], or throws exception if there is no or more than one matching
element.\n */\npublic inline fun ShortArray.single(predicate: (Short) -> Boolean): Short {\n    var single: Short? =
null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) throw
IllegalArgumentException("Array contains more than one matching element.")\n                single = element\n
found = true\n            }\n        }\n        if (!found) throw NoSuchElementException("Array contains no element matching
the predicate.")\n        @Suppress("UNCHECKED_CAST")\n        return single as Short\n}\n\n/**\n * Returns the
single element matching the given [predicate], or throws exception if there is no or more than one matching
element.\n */\npublic inline fun IntArray.single(predicate: (Int) -> Boolean): Int {\n    var single: Int? = null\n    var
found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) throw
IllegalArgumentException("Array contains more than one matching element.")\n                single = element\n
found = true\n            }\n        }\n        if (!found) throw NoSuchElementException("Array contains no element matching
the predicate.")\n        @Suppress("UNCHECKED_CAST")\n        return single as Int\n}\n\n/**\n * Returns the single
element matching the given [predicate], or throws exception if there is no or more than one matching element.\n
*/\npublic inline fun LongArray.single(predicate: (Long) -> Boolean): Long {\n    var single: Long? = null\n    var
found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) throw
IllegalArgumentException("Array contains more than one matching element.")\n                single = element\n
found = true\n            }\n        }\n        if (!found) throw NoSuchElementException("Array contains no element matching
the predicate.")\n        @Suppress("UNCHECKED_CAST")\n        return single as Long\n}\n\n/**\n * Returns the
single element matching the given [predicate], or throws exception if there is no or more than one matching
element.\n */\npublic inline fun FloatArray.single(predicate: (Float) -> Boolean): Float {\n    var single: Float? =
null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) throw
IllegalArgumentException("Array contains more than one matching element.")\n                single = element\n
found = true\n            }\n        }\n        if (!found) throw NoSuchElementException("Array contains no element matching
the predicate.")\n        @Suppress("UNCHECKED_CAST")\n        return single as Float\n}\n\n/**\n * Returns the
single element matching the given [predicate], or throws exception if there is no or more than one matching
element.\n */\npublic inline fun DoubleArray.single(predicate: (Double) -> Boolean): Double {\n    var single:
Double? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found)
throw NoSuchElementException("Array contains more than one matching element.")\n                single = element\n
found = true\n            }\n        }\n        if (!found) throw NoSuchElementException("Array contains no element
matching the predicate.")\n        @Suppress("UNCHECKED_CAST")\n        return single as Double\n}\n\n/**\n *
Returns the single element matching the given [predicate], or throws exception if there is no or more than one
matching element.\n */\npublic inline fun BooleanArray.single(predicate: (Boolean) -> Boolean): Boolean {\n    var
single: Boolean? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if
(found) throw NoSuchElementException("Array contains more than one matching element.")\n                single =
element\n            found = true\n        }\n        if (!found) throw NoSuchElementException("Array contains no
element matching the predicate.")\n        @Suppress("UNCHECKED_CAST")\n        return single as
Boolean\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no
or more than one matching element.\n */\npublic inline fun CharArray.single(predicate: (Char) -> Boolean): Char
{\n    var single: Char? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if
(found) throw NoSuchElementException("Array contains more than one matching element.")\n                single =
element\n            found = true\n        }\n        if (!found) throw NoSuchElementException("Array contains no
element matching the predicate.")\n        @Suppress("UNCHECKED_CAST")\n        return single as Char\n}\n\n/**\n
* Returns single element, or `null` if the array is empty or has more than one element.\n */\npublic fun <T>
Array<out T>.singleOrNull(): T? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns single element, or
`null` if the array is empty or has more than one element.\n */\npublic fun ByteArray.singleOrNull(): Byte? {\n

```

```

return if (size == 1) this[0] else null\n\n\n * Returns single element, or `null` if the array is empty or has more
than one element.\n *^npublic fun ShortArray.singleOrNull(): Short? {\n  return if (size == 1) this[0] else
null\n\n\n * Returns single element, or `null` if the array is empty or has more than one element.\n *^npublic
fun IntArray.singleOrNull(): Int? {\n  return if (size == 1) this[0] else null\n\n\n * Returns single element, or
`null` if the array is empty or has more than one element.\n *^npublic fun LongArray.singleOrNull(): Long? {\n
return if (size == 1) this[0] else null\n\n\n * Returns single element, or `null` if the array is empty or has more
than one element.\n *^npublic fun FloatArray.singleOrNull(): Float? {\n  return if (size == 1) this[0] else
null\n\n\n * Returns single element, or `null` if the array is empty or has more than one element.\n *^npublic
fun DoubleArray.singleOrNull(): Double? {\n  return if (size == 1) this[0] else null\n\n\n * Returns single
element, or `null` if the array is empty or has more than one element.\n *^npublic fun BooleanArray.singleOrNull():
Boolean? {\n  return if (size == 1) this[0] else null\n\n\n * Returns single element, or `null` if the array is
empty or has more than one element.\n *^npublic fun CharArray.singleOrNull(): Char? {\n  return if (size == 1)
this[0] else null\n\n\n * Returns the single element matching the given [predicate], or `null` if element was not
found or more than one element was found.\n *^npublic inline fun <T> Array<out T>.singleOrNull(predicate: (T) -
> Boolean): T? {\n  var single: T? = null\n  var found = false\n  for (element in this) {\n    if
(predicate(element)) {\n      if (found) return null\n      single = element\n      found = true\n    }\n  }\n  if (!found) return null\n  return single\n\n\n * Returns the single element matching the given [predicate], or
`null` if element was not found or more than one element was found.\n *^npublic inline fun
ByteArray.singleOrNull(predicate: (Byte) -> Boolean): Byte? {\n  var single: Byte? = null\n  var found = false\n
for (element in this) {\n    if (predicate(element)) {\n      if (found) return null\n      single = element\n
found = true\n    }\n  }\n  if (!found) return null\n  return single\n\n\n * Returns the single element
matching the given [predicate], or `null` if element was not found or more than one element was found.\n *^npublic
inline fun ShortArray.singleOrNull(predicate: (Short) -> Boolean): Short? {\n  var single: Short? = null\n  var
found = false\n  for (element in this) {\n    if (predicate(element)) {\n      if (found) return null\n      single
= element\n      found = true\n    }\n  }\n  if (!found) return null\n  return single\n\n\n * Returns the
single element matching the given [predicate], or `null` if element was not found or more than one element was
found.\n *^npublic inline fun IntArray.singleOrNull(predicate: (Int) -> Boolean): Int? {\n  var single: Int? = null\n
var found = false\n  for (element in this) {\n    if (predicate(element)) {\n      if (found) return null\n
single = element\n      found = true\n    }\n  }\n  if (!found) return null\n  return single\n\n\n *
Returns the single element matching the given [predicate], or `null` if element was not found or more than one
element was found.\n *^npublic inline fun LongArray.singleOrNull(predicate: (Long) -> Boolean): Long? {\n  var
single: Long? = null\n  var found = false\n  for (element in this) {\n    if (predicate(element)) {\n      if
(found) return null\n      single = element\n      found = true\n    }\n  }\n  if (!found) return null\n  return
single\n\n\n * Returns the single element matching the given [predicate], or `null` if element was not found or
more than one element was found.\n *^npublic inline fun FloatArray.singleOrNull(predicate: (Float) -> Boolean):
Float? {\n  var single: Float? = null\n  var found = false\n  for (element in this) {\n    if (predicate(element))
{\n      if (found) return null\n      single = element\n      found = true\n    }\n  }\n  if (!found) return
null\n  return single\n\n\n * Returns the single element matching the given [predicate], or `null` if element
was not found or more than one element was found.\n *^npublic inline fun DoubleArray.singleOrNull(predicate:
(Double) -> Boolean): Double? {\n  var single: Double? = null\n  var found = false\n  for (element in this) {\n
if (predicate(element)) {\n      if (found) return null\n      single = element\n      found = true\n    }\n
}\n  if (!found) return null\n  return single\n\n\n * Returns the single element matching the given
[predicate], or `null` if element was not found or more than one element was found.\n *^npublic inline fun
BooleanArray.singleOrNull(predicate: (Boolean) -> Boolean): Boolean? {\n  var single: Boolean? = null\n  var
found = false\n  for (element in this) {\n    if (predicate(element)) {\n      if (found) return null\n      single
= element\n      found = true\n    }\n  }\n  if (!found) return null\n  return single\n\n\n * Returns the
single element matching the given [predicate], or `null` if element was not found or more than one element was
found.\n *^npublic inline fun CharArray.singleOrNull(predicate: (Char) -> Boolean): Char? {\n  var single: Char?

```

```

= null\n  var found = false\n  for (element in this) {\n    if (predicate(element)) {\n      if (found) return\n      null\n      single = element\n      found = true\n    }\n  }\n  if (!found) return null\n  return\n  single\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun <T> Array<out T>.drop(n: Int): List<T> {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return takeLast((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun ByteArray.drop(n: Int): List<Byte> {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return takeLast((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun ShortArray.drop(n: Int): List<Short> {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return takeLast((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun IntArray.drop(n: Int): List<Int> {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return takeLast((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun LongArray.drop(n: Int): List<Long> {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return takeLast((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun FloatArray.drop(n: Int): List<Float> {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return takeLast((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun DoubleArray.drop(n: Int): List<Double> {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return takeLast((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun BooleanArray.drop(n: Int): List<Boolean>\n {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return takeLast((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except first [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun CharArray.drop(n: Int): List<Char> {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return takeLast((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun <T> Array<out T>.dropLast(n: Int): List<T>\n {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return take((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample\n * samples.collections.Collections.Transformations.drop\n *\npublic fun ByteArray.dropLast(n: Int): List<Byte> {\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  return take((size -\n  n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws\n * IllegalArgumentException if [n] is negative.\n * \n * @sample

```

```

samples.collections.Collections.Transformations.drop\n *\npublic fun ShortArray.dropLast(n: Int): List<Short> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return take((size -
n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws
IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic fun IntArray.dropLast(n: Int): List<Int> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return take((size -
n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws
IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic fun LongArray.dropLast(n: Int): List<Long> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return take((size -
n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws
IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic fun FloatArray.dropLast(n: Int): List<Float> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return take((size -
n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws
IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic fun DoubleArray.dropLast(n: Int): List<Double>
{\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return take((size -
n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws
IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic fun BooleanArray.dropLast(n: Int):
List<Boolean> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return take((size -
n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last [n] elements.\n * \n * @throws
IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic fun CharArray.dropLast(n: Int): List<Char> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n return take((size -
n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given
[predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic inline fun <T>
Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {\n for (index in lastIndex downTo 0) {\n if
(!predicate(this[index])) {\n return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns
a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic inline fun ByteArray.dropLastWhile(predicate:
(Byte) -> Boolean): List<Byte> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n
return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements
except last elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic inline fun ShortArray.dropLastWhile(predicate:
(Short) -> Boolean): List<Short> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n
return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements
except last elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic inline fun IntArray.dropLastWhile(predicate:
(Int) -> Boolean): List<Int> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n
return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements
except last elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic inline fun LongArray.dropLastWhile(predicate:
(Long) -> Boolean): List<Long> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n
return take(index + 1)\n }\n }\n return emptyList()\n}\n\n/**\n * Returns a list containing all elements
except last elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic inline fun FloatArray.dropLastWhile(predicate:

```

```

(Float) -> Boolean): List<Float> {\n  for (index in lastIndex downTo 0) {\n    if (!predicate(this[index])) {\n      return take(index + 1)\n    }\n  }\n  return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun DoubleArray.dropLastWhile(predicate: (Double) -> Boolean): List<Double> {\n  for (index in lastIndex downTo 0) {\n    if (!predicate(this[index])) {\n      return take(index + 1)\n    }\n  }\n  return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun BooleanArray.dropLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {\n  for (index in lastIndex downTo 0) {\n    if (!predicate(this[index])) {\n      return take(index + 1)\n    }\n  }\n  return emptyList()\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun CharArray.dropLastWhile(predicate: (Char) -> Boolean): List<Char> {\n  for (index in lastIndex downTo 0) {\n    if (!predicate(this[index])) {\n      return take(index + 1)\n    }\n  }\n  return emptyList()\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun <T> Array<out T>.dropWhile(predicate: (T) -> Boolean): List<T> {\n  var yielding = false\n  val list = ArrayList<T>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item))\n      list.add(item)\n      yielding = true\n  }\n  return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun ByteArray.dropWhile(predicate: (Byte) -> Boolean): List<Byte> {\n  var yielding = false\n  val list = ArrayList<Byte>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item))\n      list.add(item)\n      yielding = true\n  }\n  return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun ShortArray.dropWhile(predicate: (Short) -> Boolean): List<Short> {\n  var yielding = false\n  val list = ArrayList<Short>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item))\n      list.add(item)\n      yielding = true\n  }\n  return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun IntArray.dropWhile(predicate: (Int) -> Boolean): List<Int> {\n  var yielding = false\n  val list = ArrayList<Int>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item))\n      list.add(item)\n      yielding = true\n  }\n  return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun LongArray.dropWhile(predicate: (Long) -> Boolean): List<Long> {\n  var yielding = false\n  val list = ArrayList<Long>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item))\n      list.add(item)\n      yielding = true\n  }\n  return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun FloatArray.dropWhile(predicate: (Float) -> Boolean): List<Float> {\n  var yielding = false\n  val list = ArrayList<Float>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item))\n      list.add(item)\n      yielding = true\n  }\n  return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun DoubleArray.dropWhile(predicate: (Double) -> Boolean): List<Double> {\n  var yielding = false\n  val list = ArrayList<Double>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item))\n      list.add(item)\n      yielding = true\n  }\n  return list\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.drop\n */\npublic inline fun BooleanArray.dropWhile(predicate: (Boolean) -> Boolean): List<Boolean> {\n  var yielding = false\n}

```



```

val list = ArrayList<Boolean>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if\n      (!predicate(item)) {\n        list.add(item)\n        yielding = true\n      }\n  return list\n}\n\n/**\n * Returns a list\n * containing all elements except first elements that satisfy the given [predicate].\n * @sample\n * samples.collections.Collections.Transformations.drop\n */\npublic inline fun CharArray.dropWhile(predicate:\n (Char) -> Boolean): List<Char> {\n  var yielding = false\n  val list = ArrayList<Char>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item)) {\n      list.add(item)\n      yielding =\n true\n    }\n  return list\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n */\npublic inline fun <T> Array<out T>.filter(predicate: (T) -> Boolean): List<T> {\n  return\n filterTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given\n [predicate].\n */\npublic inline fun ByteArray.filter(predicate: (Byte) -> Boolean): List<Byte> {\n  return\n filterTo(ArrayList<Byte>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given\n [predicate].\n */\npublic inline fun ShortArray.filter(predicate: (Short) -> Boolean): List<Short> {\n  return\n filterTo(ArrayList<Short>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given\n [predicate].\n */\npublic inline fun IntArray.filter(predicate: (Int) -> Boolean): List<Int> {\n  return\n filterTo(ArrayList<Int>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given\n [predicate].\n */\npublic inline fun LongArray.filter(predicate: (Long) -> Boolean): List<Long> {\n  return\n filterTo(ArrayList<Long>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given\n [predicate].\n */\npublic inline fun FloatArray.filter(predicate: (Float) -> Boolean): List<Float> {\n  return\n filterTo(ArrayList<Float>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given\n [predicate].\n */\npublic inline fun DoubleArray.filter(predicate: (Double) -> Boolean): List<Double> {\n  return\n filterTo(ArrayList<Double>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given\n [predicate].\n */\npublic inline fun BooleanArray.filter(predicate: (Boolean) -> Boolean): List<Boolean> {\n  return\n filterTo(ArrayList<Boolean>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given\n [predicate].\n */\npublic inline fun CharArray.filter(predicate: (Char) -> Boolean): List<Char> {\n  return\n filterTo(ArrayList<Char>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given\n [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns\n the result of predicate evaluation on the element.\n */\npublic inline fun <T> Array<out T>.filterIndexed(predicate:\n (index: Int, T) -> Boolean): List<T> {\n  return filterIndexedTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Returns a\n list containing only elements matching the given [predicate].\n * @param [predicate] function that takes the index of\n an element and the element itself\n * and returns the result of predicate evaluation on the element.\n */\npublic inline\n fun ByteArray.filterIndexed(predicate: (index: Int, Byte) -> Boolean): List<Byte> {\n  return\n filterIndexedTo(ArrayList<Byte>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the\n given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and\n returns the result of predicate evaluation on the element.\n */\npublic inline fun ShortArray.filterIndexed(predicate:\n (index: Int, Short) -> Boolean): List<Short> {\n  return filterIndexedTo(ArrayList<Short>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes\n the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n */\npublic inline fun IntArray.filterIndexed(predicate: (index: Int, Int) -> Boolean): List<Int> {\n  return\n filterIndexedTo(ArrayList<Int>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the\n given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and\n returns the result of predicate evaluation on the element.\n */\npublic inline fun LongArray.filterIndexed(predicate:\n (index: Int, Long) -> Boolean): List<Long> {\n  return filterIndexedTo(ArrayList<Long>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param [predicate] function that takes\n the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n */\npublic inline fun FloatArray.filterIndexed(predicate: (index: Int, Float) -> Boolean): List<Float> {\n  return\n filterIndexedTo(ArrayList<Float>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the\n given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and\n returns the result of predicate evaluation on the element.\n */\npublic inline fun

```

```

DoubleArray.filterIndexed(predicate: (index: Int, Double) -> Boolean): List<Double> {\n  return
filterIndexedTo(ArrayList<Double>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the
given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and
returns the result of predicate evaluation on the element.\n */\npublic inline fun
BooleanArray.filterIndexed(predicate: (index: Int, Boolean) -> Boolean): List<Boolean> {\n  return
filterIndexedTo(ArrayList<Boolean>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching
the given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n *
and returns the result of predicate evaluation on the element.\n */\npublic inline fun
CharArray.filterIndexed(predicate: (index: Int, Char) -> Boolean): List<Char> {\n  return
filterIndexedTo(ArrayList<Char>(), predicate)\n}\n\n/**\n * Appends all elements matching the given [predicate] to
the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n *
and returns the result of predicate evaluation on the element.\n */\npublic inline fun <T, C : MutableCollection<in
T>> Array<out T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {\n  forEachIndexed {
index, element ->\n    if (predicate(index, element)) destination.add(element)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n *
@param [predicate] function that takes the index of an element and the element itself\n * and returns the result of
predicate evaluation on the element.\n */\npublic inline fun <C : MutableCollection<in Byte>>
ByteArray.filterIndexedTo(destination: C, predicate: (index: Int, Byte) -> Boolean): C {\n  forEachIndexed {
index, element ->\n    if (predicate(index, element)) destination.add(element)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n *
@param [predicate] function that takes the index of an element and the element itself\n * and returns the result of
predicate evaluation on the element.\n */\npublic inline fun <C : MutableCollection<in Short>>
ShortArray.filterIndexedTo(destination: C, predicate: (index: Int, Short) -> Boolean): C {\n  forEachIndexed {
index, element ->\n    if (predicate(index, element)) destination.add(element)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n *
@param [predicate] function that takes the index of an element and the element itself\n * and returns the result of
predicate evaluation on the element.\n */\npublic inline fun <C : MutableCollection<in Int>>
IntArray.filterIndexedTo(destination: C, predicate: (index: Int, Int) -> Boolean): C {\n  forEachIndexed { index,
element ->\n    if (predicate(index, element)) destination.add(element)\n  }\n  return destination\n}\n\n/**\n *
Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that
takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the
element.\n */\npublic inline fun <C : MutableCollection<in Long>> LongArray.filterIndexedTo(destination: C,
predicate: (index: Int, Long) -> Boolean): C {\n  forEachIndexed { index, element ->\n    if (predicate(index,
element)) destination.add(element)\n  }\n  return destination\n}\n\n/**\n * Appends all elements matching the
given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and
the element itself\n * and returns the result of predicate evaluation on the element.\n */\npublic inline fun <C :
MutableCollection<in Float>> FloatArray.filterIndexedTo(destination: C, predicate: (index: Int, Float) -> Boolean):
C {\n  forEachIndexed { index, element ->\n    if (predicate(index, element)) destination.add(element)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n *
@param [predicate] function that takes the index of an element and the element itself\n * and returns the result of
predicate evaluation on the element.\n */\npublic inline fun <C : MutableCollection<in Double>>
DoubleArray.filterIndexedTo(destination: C, predicate: (index: Int, Double) -> Boolean): C {\n  forEachIndexed {
index, element ->\n    if (predicate(index, element)) destination.add(element)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n *
@param [predicate] function that takes the index of an element and the element itself\n * and returns the result of
predicate evaluation on the element.\n */\npublic inline fun <C : MutableCollection<in Boolean>>
BooleanArray.filterIndexedTo(destination: C, predicate: (index: Int, Boolean) -> Boolean): C {\n  forEachIndexed {
index, element ->\n    if (predicate(index, element)) destination.add(element)\n  }\n  return

```

```

destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n *
@param [predicate] function that takes the index of an element and the element itself\n * and returns the result of
predicate evaluation on the element.\n *\npublic inline fun <C : MutableCollection<in Char>>
CharArray.filterIndexedTo(destination: C, predicate: (index: Int, Char) -> Boolean): C {\n  forEachIndexed {
index, element ->\n    if (predicate(index, element)) destination.add(element)\n  }\n  return
destination\n}\n\n/**\n * Returns a list containing all elements that are instances of specified type parameter R.\n *
*\npublic inline fun <reified R> Array<*>.filterIsInstance(): List<@kotlin.internal.NoInfer R> {\n  return
filterIsInstanceTo(ArrayList<R>())\n}\n\n/**\n * Appends all elements that are instances of specified type
parameter R to the given [destination].\n *\npublic inline fun <reified R, C : MutableCollection<in R>>
Array<*>.filterIsInstanceTo(destination: C): C {\n  for (element in this) if (element is R)
destination.add(element)\n  return destination\n}\n\n/**\n * Returns a list containing all elements not matching the
given [predicate].\n *\npublic inline fun <T> Array<out T>.filterNot(predicate: (T) -> Boolean): List<T> {\n
return filterNotTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the
given [predicate].\n *\npublic inline fun ByteArray.filterNot(predicate: (Byte) -> Boolean): List<Byte> {\n
return filterNotTo(ArrayList<Byte>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the
given [predicate].\n *\npublic inline fun ShortArray.filterNot(predicate: (Short) -> Boolean): List<Short> {\n
return filterNotTo(ArrayList<Short>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching
the given [predicate].\n *\npublic inline fun IntArray.filterNot(predicate: (Int) -> Boolean): List<Int> {\n
return filterNotTo(ArrayList<Int>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given
[predicate].\n *\npublic inline fun LongArray.filterNot(predicate: (Long) -> Boolean): List<Long> {\n
return filterNotTo(ArrayList<Long>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the
given [predicate].\n *\npublic inline fun FloatArray.filterNot(predicate: (Float) -> Boolean): List<Float> {\n
return filterNotTo(ArrayList<Float>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching
the given [predicate].\n *\npublic inline fun DoubleArray.filterNot(predicate: (Double) -> Boolean): List<Double>
{\n  return filterNotTo(ArrayList<Double>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not
matching the given [predicate].\n *\npublic inline fun BooleanArray.filterNot(predicate: (Boolean) -> Boolean):
List<Boolean> {\n  return filterNotTo(ArrayList<Boolean>(), predicate)\n}\n\n/**\n * Returns a list containing all
elements not matching the given [predicate].\n *\npublic inline fun CharArray.filterNot(predicate: (Char) ->
Boolean): List<Char> {\n  return filterNotTo(ArrayList<Char>(), predicate)\n}\n\n/**\n * Returns a list containing
all elements that are not `null`.\n *\npublic fun <T : Any> Array<out T?>.filterNotNull(): List<T> {\n  return
filterNotNullTo(ArrayList<T>())\n}\n\n/**\n * Appends all elements that are not `null` to the given [destination].\n *
*\npublic fun <C : MutableCollection<in T>, T : Any> Array<out T?>.filterNotNullTo(destination: C): C {\n  for
(element in this) if (element != null) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements not matching the given [predicate] to the given [destination].\n *\npublic inline fun <T, C :
MutableCollection<in T>> Array<out T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {\n  for
(element in this) if (!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements not matching the given [predicate] to the given [destination].\n *\npublic inline fun <C :
MutableCollection<in Byte>> ByteArray.filterNotTo(destination: C, predicate: (Byte) -> Boolean): C {\n  for
(element in this) if (!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements not matching the given [predicate] to the given [destination].\n *\npublic inline fun <C :
MutableCollection<in Short>> ShortArray.filterNotTo(destination: C, predicate: (Short) -> Boolean): C {\n  for
(element in this) if (!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements not matching the given [predicate] to the given [destination].\n *\npublic inline fun <C :
MutableCollection<in Int>> IntArray.filterNotTo(destination: C, predicate: (Int) -> Boolean): C {\n  for (element
in this) if (!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all elements
not matching the given [predicate] to the given [destination].\n *\npublic inline fun <C : MutableCollection<in
Long>> LongArray.filterNotTo(destination: C, predicate: (Long) -> Boolean): C {\n  for (element in this) if
(!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all elements not

```

```

matching the given [predicate] to the given [destination].\n *\npublic inline fun <C : MutableCollection<in Float>>
FloatArray.filterNotTo(destination: C, predicate: (Float) -> Boolean): C {\n  for (element in this) if
(!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all elements not
matching the given [predicate] to the given [destination].\n *\npublic inline fun <C : MutableCollection<in
Double>> DoubleArray.filterNotTo(destination: C, predicate: (Double) -> Boolean): C {\n  for (element in this) if
(!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all elements not
matching the given [predicate] to the given [destination].\n *\npublic inline fun <C : MutableCollection<in
Boolean>> BooleanArray.filterNotTo(destination: C, predicate: (Boolean) -> Boolean): C {\n  for (element in this)
if (!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all elements not
matching the given [predicate] to the given [destination].\n *\npublic inline fun <C : MutableCollection<in Char>>
CharArray.filterNotTo(destination: C, predicate: (Char) -> Boolean): C {\n  for (element in this) if
(!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all elements matching
the given [predicate] to the given [destination].\n *\npublic inline fun <T, C : MutableCollection<in T>> Array<out
T>.filterTo(destination: C, predicate: (T) -> Boolean): C {\n  for (element in this) if (predicate(element))
destination.add(element)\n  return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to
the given [destination].\n *\npublic inline fun <C : MutableCollection<in Byte>> ByteArray.filterTo(destination: C,
predicate: (Byte) -> Boolean): C {\n  for (element in this) if (predicate(element)) destination.add(element)\n
return destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n
*\npublic inline fun <C : MutableCollection<in Short>> ShortArray.filterTo(destination: C, predicate: (Short) ->
Boolean): C {\n  for (element in this) if (predicate(element)) destination.add(element)\n  return
destination\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n
*\npublic inline fun <C : MutableCollection<in Int>> IntArray.filterTo(destination: C, predicate: (Int) -> Boolean):
C {\n  for (element in this) if (predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n *
Appends all elements matching the given [predicate] to the given [destination].\n *\npublic inline fun <C :
MutableCollection<in Long>> LongArray.filterTo(destination: C, predicate: (Long) -> Boolean): C {\n  for
(element in this) if (predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements matching the given [predicate] to the given [destination].\n *\npublic inline fun <C :
MutableCollection<in Float>> FloatArray.filterTo(destination: C, predicate: (Float) -> Boolean): C {\n  for
(element in this) if (predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements matching the given [predicate] to the given [destination].\n *\npublic inline fun <C :
MutableCollection<in Double>> DoubleArray.filterTo(destination: C, predicate: (Double) -> Boolean): C {\n  for
(element in this) if (predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements matching the given [predicate] to the given [destination].\n *\npublic inline fun <C :
MutableCollection<in Boolean>> BooleanArray.filterTo(destination: C, predicate: (Boolean) -> Boolean): C {\n
for (element in this) if (predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends
all elements matching the given [predicate] to the given [destination].\n *\npublic inline fun <C :
MutableCollection<in Char>> CharArray.filterTo(destination: C, predicate: (Char) -> Boolean): C {\n  for
(element in this) if (predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Returns a list
containing elements at indices in the specified [indices] range.\n *\npublic fun <T> Array<out T>.slice(indices:
IntRange): List<T> {\n  if (indices.isEmpty()) return listOf()\n  return copyOfRange(indices.start,
indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified
[indices] range.\n *\npublic fun ByteArray.slice(indices: IntRange): List<Byte> {\n  if (indices.isEmpty()) return
listOf()\n  return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list
containing elements at indices in the specified [indices] range.\n *\npublic fun ShortArray.slice(indices: IntRange):
List<Short> {\n  if (indices.isEmpty()) return listOf()\n  return copyOfRange(indices.start, indices.endInclusive +
1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n *\npublic
fun IntArray.slice(indices: IntRange): List<Int> {\n  if (indices.isEmpty()) return listOf()\n  return
copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at

```

```

indices in the specified [indices] range.\n */\npublic fun LongArray.slice(indices: IntRange): List<Long> {\n if
(indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive +
1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic
fun FloatArray.slice(indices: IntRange): List<Float> {\n if (indices.isEmpty()) return listOf()\n return
copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at
indices in the specified [indices] range.\n */\npublic fun DoubleArray.slice(indices: IntRange): List<Double> {\n
if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive +
1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n */\npublic
fun BooleanArray.slice(indices: IntRange): List<Boolean> {\n if (indices.isEmpty()) return listOf()\n return
copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n\n/**\n * Returns a list containing elements at
indices in the specified [indices] range.\n */\npublic fun CharArray.slice(indices: IntRange): List<Char> {\n if
(indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive +
1).asList()\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun <T> Array<out
T>.slice(indices: Iterable<Int>): List<T> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0)
return emptyList()\n val list = ArrayList<T>(size)\n for (index in indices) {\n list.add(get(index))\n }\n
return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun
ByteArray.slice(indices: Iterable<Int>): List<Byte> {\n val size = indices.collectionSizeOrDefault(10)\n if (size
== 0) return emptyList()\n val list = ArrayList<Byte>(size)\n for (index in indices) {\n list.add(get(index))\n
}\n return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n */\npublic fun
ShortArray.slice(indices: Iterable<Int>): List<Short> {\n val size = indices.collectionSizeOrDefault(10)\n if
(size == 0) return emptyList()\n val list = ArrayList<Short>(size)\n for (index in indices) {\n
list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n
*/\npublic fun IntArray.slice(indices: Iterable<Int>): List<Int> {\n val size =
indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<Int>(size)\n for
(index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing elements at
specified [indices].\n */\npublic fun LongArray.slice(indices: Iterable<Int>): List<Long> {\n val size =
indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<Long>(size)\n
for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing
elements at specified [indices].\n */\npublic fun FloatArray.slice(indices: Iterable<Int>): List<Float> {\n val size =
indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list = ArrayList<Float>(size)\n
for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n * Returns a list containing
elements at specified [indices].\n */\npublic fun DoubleArray.slice(indices: Iterable<Int>): List<Double> {\n val
size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list =
ArrayList<Double>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n *
Returns a list containing elements at specified [indices].\n */\npublic fun BooleanArray.slice(indices: Iterable<Int>):
List<Boolean> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list
= ArrayList<Boolean>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n *
Returns a list containing elements at specified [indices].\n */\npublic fun CharArray.slice(indices: Iterable<Int>):
List<Char> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list =
ArrayList<Char>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n *
Returns an array containing elements of this array at specified [indices].\n */\npublic fun <T>
Array<T>.sliceArray(indices: Collection<Int>): Array<T> {\n val result = arrayOfNulls(this, indices.size)\n var
targetIndex = 0\n for (sourceIndex in indices) {\n result[targetIndex++] = this[sourceIndex]\n }\n return
result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun
ByteArray.sliceArray(indices: Collection<Int>): ByteArray {\n val result = ByteArray(indices.size)\n var
targetIndex = 0\n for (sourceIndex in indices) {\n result[targetIndex++] = this[sourceIndex]\n }\n return
result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n */\npublic fun
ShortArray.sliceArray(indices: Collection<Int>): ShortArray {\n val result = ShortArray(indices.size)\n var

```

```

targetIndex = 0\n  for (sourceIndex in indices) {\n    result[targetIndex++] = this[sourceIndex]\n  }\n  return
result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n *\npublic fun
IntArray.sliceArray(indices: Collection<Int>): IntArray {\n  val result = IntArray(indices.size)\n  var targetIndex
= 0\n  for (sourceIndex in indices) {\n    result[targetIndex++] = this[sourceIndex]\n  }\n  return
result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n *\npublic fun
LongArray.sliceArray(indices: Collection<Int>): LongArray {\n  val result = LongArray(indices.size)\n  var
targetIndex = 0\n  for (sourceIndex in indices) {\n    result[targetIndex++] = this[sourceIndex]\n  }\n  return
result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n *\npublic fun
FloatArray.sliceArray(indices: Collection<Int>): FloatArray {\n  val result = FloatArray(indices.size)\n  var
targetIndex = 0\n  for (sourceIndex in indices) {\n    result[targetIndex++] = this[sourceIndex]\n  }\n  return
result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n *\npublic fun
DoubleArray.sliceArray(indices: Collection<Int>): DoubleArray {\n  val result = DoubleArray(indices.size)\n
var targetIndex = 0\n  for (sourceIndex in indices) {\n    result[targetIndex++] = this[sourceIndex]\n  }\n
return result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n *\npublic fun
BooleanArray.sliceArray(indices: Collection<Int>): BooleanArray {\n  val result = BooleanArray(indices.size)\n
var targetIndex = 0\n  for (sourceIndex in indices) {\n    result[targetIndex++] = this[sourceIndex]\n  }\n
return result\n}\n\n/**\n * Returns an array containing elements of this array at specified [indices].\n *\npublic fun
CharArray.sliceArray(indices: Collection<Int>): CharArray {\n  val result = CharArray(indices.size)\n  var
targetIndex = 0\n  for (sourceIndex in indices) {\n    result[targetIndex++] = this[sourceIndex]\n  }\n  return
result\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n *\npublic fun
<T> Array<T>.sliceArray(indices: IntRange): Array<T> {\n  if (indices.isEmpty()) return copyOfRange(0, 0)\n
return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at
indices in the specified [indices] range.\n *\npublic fun ByteArray.sliceArray(indices: IntRange): ByteArray {\n  if
(indices.isEmpty()) return ByteArray(0)\n  return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n
* Returns an array containing elements at indices in the specified [indices] range.\n *\npublic fun
ShortArray.sliceArray(indices: IntRange): ShortArray {\n  if (indices.isEmpty()) return ShortArray(0)\n  return
copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at indices
in the specified [indices] range.\n *\npublic fun IntArray.sliceArray(indices: IntRange): IntArray {\n  if
(indices.isEmpty()) return IntArray(0)\n  return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n
* Returns an array containing elements at indices in the specified [indices] range.\n *\npublic fun
LongArray.sliceArray(indices: IntRange): LongArray {\n  if (indices.isEmpty()) return LongArray(0)\n  return
copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at indices
in the specified [indices] range.\n *\npublic fun FloatArray.sliceArray(indices: IntRange): FloatArray {\n  if
(indices.isEmpty()) return FloatArray(0)\n  return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n
* Returns an array containing elements at indices in the specified [indices] range.\n *\npublic fun
DoubleArray.sliceArray(indices: IntRange): DoubleArray {\n  if (indices.isEmpty()) return DoubleArray(0)\n
return copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns an array containing elements at
indices in the specified [indices] range.\n *\npublic fun BooleanArray.sliceArray(indices: IntRange): BooleanArray
{\n  if (indices.isEmpty()) return BooleanArray(0)\n  return copyOfRange(indices.start, indices.endInclusive +
1)\n}\n\n/**\n * Returns an array containing elements at indices in the specified [indices] range.\n *\npublic fun
CharArray.sliceArray(indices: IntRange): CharArray {\n  if (indices.isEmpty()) return CharArray(0)\n  return
copyOfRange(indices.start, indices.endInclusive + 1)\n}\n\n/**\n * Returns a list containing first [n] elements.\n *\n
\n * @throws IllegalArgumentException if [n] is negative.\n *\n\n * @sample
samples.collections.Collections.Transformations.take\n *\npublic fun <T> Array<out T>.take(n: Int): List<T> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  if (n == 0) return emptyList()\n  if (n >=
size) return toList()\n  if (n == 1) return listOf(this[0])\n  var count = 0\n  val list = ArrayList<T>(n)\n  for
(item in this) {\n    if (count++ == n)\n      break\n    list.add(item)\n  }\n  return list\n}\n\n/**\n * Returns
a list containing first [n] elements.\n *\n\n * @throws IllegalArgumentException if [n] is negative.\n *\n\n * @sample

```

```

samples.collections.Collections.Transformations.take\n *^\npublic fun ByteArray.take(n: Int): List<Byte> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n if (n >=
size) return toList()\n if (n == 1) return listOf(this[0])\n var count = 0\n val list = ArrayList<Byte>(n)\n for
(item in this) {\n if (count++ == n)\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns
a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n *^\npublic fun ShortArray.take(n: Int): List<Short> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n if (n >=
size) return toList()\n if (n == 1) return listOf(this[0])\n var count = 0\n val list = ArrayList<Short>(n)\n for
(item in this) {\n if (count++ == n)\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns
a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n *^\npublic fun IntArray.take(n: Int): List<Int> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n if (n >=
size) return toList()\n if (n == 1) return listOf(this[0])\n var count = 0\n val list = ArrayList<Int>(n)\n for
(item in this) {\n if (count++ == n)\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns
a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n *^\npublic fun LongArray.take(n: Int): List<Long> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n if (n >=
size) return toList()\n if (n == 1) return listOf(this[0])\n var count = 0\n val list = ArrayList<Long>(n)\n for
(item in this) {\n if (count++ == n)\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns
a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n *^\npublic fun FloatArray.take(n: Int): List<Float> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n if (n >=
size) return toList()\n if (n == 1) return listOf(this[0])\n var count = 0\n val list = ArrayList<Float>(n)\n for
(item in this) {\n if (count++ == n)\n break\n list.add(item)\n }\n return list\n}\n\n/**\n * Returns
a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n *^\npublic fun DoubleArray.take(n: Int): List<Double> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n if (n >=
size) return toList()\n if (n == 1) return listOf(this[0])\n var count = 0\n val list = ArrayList<Double>(n)\n
for (item in this) {\n if (count++ == n)\n break\n list.add(item)\n }\n return list\n}\n\n/**\n *
Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n *
@sample samples.collections.Collections.Transformations.take\n *^\npublic fun BooleanArray.take(n: Int):
List<Boolean> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return
emptyList()\n if (n >= size) return toList()\n if (n == 1) return listOf(this[0])\n var count = 0\n val list =
ArrayList<Boolean>(n)\n for (item in this) {\n if (count++ == n)\n break\n list.add(item)\n }\n
return list\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if
[n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.take\n *^\npublic fun
CharArray.take(n: Int): List<Char> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if
(n == 0) return emptyList()\n if (n >= size) return toList()\n if (n == 1) return listOf(this[0])\n var count = 0\n
val list = ArrayList<Char>(n)\n for (item in this) {\n if (count++ == n)\n break\n list.add(item)\n
}\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws
IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n *^\npublic fun <T> Array<out T>.takeLast(n: Int): List<T>
{\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val
size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list =
ArrayList<T>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n *
Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n *
@sample samples.collections.Collections.Transformations.take\n *^\npublic fun ByteArray.takeLast(n: Int): List<Byte> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size =

```

```

size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Byte>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last\n [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample\n samples.collections.Collections.Transformations.take\n */\npublic fun ShortArray.takeLast(n: Int): List<Short> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size =\n size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Short>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last\n [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample\n samples.collections.Collections.Transformations.take\n */\npublic fun IntArray.takeLast(n: Int): List<Int> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size =\n size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Int>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last\n [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample\n samples.collections.Collections.Transformations.take\n */\npublic fun LongArray.takeLast(n: Int): List<Long> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size =\n size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Long>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last\n [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample\n samples.collections.Collections.Transformations.take\n */\npublic fun FloatArray.takeLast(n: Int): List<Float> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val size =\n size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Float>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last\n [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample\n samples.collections.Collections.Transformations.take\n */\npublic fun DoubleArray.takeLast(n: Int): List<Double>\n {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return emptyList()\n val\n size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list =\n ArrayList<Double>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * \n * @sample\n samples.collections.Collections.Transformations.take\n */\npublic fun BooleanArray.takeLast(n: Int):\n List<Boolean> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return\n emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list\n = ArrayList<Boolean>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * \n * @sample\n samples.collections.Collections.Transformations.take\n */\npublic fun CharArray.takeLast(n: Int):\n List<Char> {\n require(n >= 0) { \"Requested element count $n is less than zero.\" }\n if (n == 0) return\n emptyList()\n val size = size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list\n = ArrayList<Char>(n)\n for (index in size - n until size)\n list.add(this[index])\n return list\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.take\n */\npublic inline fun <T> Array<out\n T>.takeLastWhile(predicate: (T) -> Boolean): List<T> {\n for (index in lastIndex downTo 0) {\n if\n (!predicate(this[index])) {\n return drop(index + 1)\n }\n }\n return toList()\n}\n\n/**\n * Returns a\n list containing last elements satisfying the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.take\n */\npublic inline fun ByteArray.takeLastWhile(predicate:\n (Byte) -> Boolean): List<Byte> {\n for (index in lastIndex downTo 0) {\n if (!predicate(this[index])) {\n\n return drop(index + 1)\n }\n }\n return toList()\n}\n\n/**\n * Returns a list containing last elements\n satisfying the given [predicate].\n * \n * @sample\n samples.collections.Collections.Transformations.take\n */\npublic\n inline fun ShortArray.takeLastWhile(predicate: (Short) -> Boolean): List<Short> {\n for (index in lastIndex\n downTo 0) {\n if (!predicate(this[index])) {\n\n return drop(index + 1)\n }\n }\n return

```



```

toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n *\npublic inline fun IntArray.takeLastWhile(predicate: (Int)
-> Boolean): List<Int> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return
drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the
given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n *\npublic inline fun
LongArray.takeLastWhile(predicate: (Long) -> Boolean): List<Long> {\n    for (index in lastIndex downTo 0) {\n
        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns
a list containing last elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n *\npublic inline fun FloatArray.takeLastWhile(predicate:
(Float) -> Boolean): List<Float> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n
        return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing last elements
satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n *\npublic
inline fun DoubleArray.takeLastWhile(predicate: (Double) -> Boolean): List<Double> {\n    for (index in lastIndex
downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return
toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n *\npublic inline fun
BooleanArray.takeLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {\n    for (index in lastIndex
downTo 0) {\n        if (!predicate(this[index])) {\n            return drop(index + 1)\n        }\n    }\n    return
toList()\n}\n\n/**\n * Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n *\npublic inline fun CharArray.takeLastWhile(predicate:
(Char) -> Boolean): List<Char> {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n
        return drop(index + 1)\n        }\n    }\n    return toList()\n}\n\n/**\n * Returns a list containing first elements
satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n *\npublic
inline fun <T> Array<out T>.takeWhile(predicate: (T) -> Boolean): List<T> {\n    val list = ArrayList<T>()\n    for
(item in this) {\n        if (!predicate(item))\n            break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n *
Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n *\npublic inline fun ByteArray.takeWhile(predicate: (Byte)
-> Boolean): List<Byte> {\n    val list = ArrayList<Byte>()\n    for (item in this) {\n        if (!predicate(item))\n
        break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n * Returns a list containing first elements satisfying the
given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n *\npublic inline fun
ShortArray.takeWhile(predicate: (Short) -> Boolean): List<Short> {\n    val list = ArrayList<Short>()\n    for (item
in this) {\n        if (!predicate(item))\n            break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n * Returns a
list containing first elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n *\npublic inline fun IntArray.takeWhile(predicate: (Int) ->
Boolean): List<Int> {\n    val list = ArrayList<Int>()\n    for (item in this) {\n        if (!predicate(item))\n
        break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n * Returns a list containing first elements satisfying the
given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n *\npublic inline fun
LongArray.takeWhile(predicate: (Long) -> Boolean): List<Long> {\n    val list = ArrayList<Long>()\n    for (item
in this) {\n        if (!predicate(item))\n            break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n * Returns a
list containing first elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n *\npublic inline fun FloatArray.takeWhile(predicate:
(Float) -> Boolean): List<Float> {\n    val list = ArrayList<Float>()\n    for (item in this) {\n        if
(!predicate(item))\n            break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n * Returns a list containing first
elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n
*\npublic inline fun DoubleArray.takeWhile(predicate: (Double) -> Boolean): List<Double> {\n    val list =
ArrayList<Double>()\n    for (item in this) {\n        if (!predicate(item))\n            break\n        list.add(item)\n    }\n
return list\n}\n\n/**\n * Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n *\npublic inline fun BooleanArray.takeWhile(predicate:

```

```

(Boolean) -> Boolean): List<Boolean> {\n  val list = ArrayList<Boolean>()\n  for (item in this) {\n    if
(!predicate(item))\n      break\n    list.add(item)\n  }\n  return list\n}\n\n/**\n * Returns a list containing first
elements satisfying the given [predicate].\n */\n * @sample samples.collections.Collections.Transformations.take\n
*/\npublic inline fun CharArray.takeWhile(predicate: (Char) -> Boolean): List<Char> {\n  val list =
ArrayList<Char>()\n  for (item in this) {\n    if (!predicate(item))\n      break\n    list.add(item)\n  }\n
return list\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n\npublic fun <T> Array<T>.reverse(): Unit {\n
val midPoint = (size / 2) - 1\n  if (midPoint < 0) return\n  var reverseIndex = lastIndex\n  for (index in
0..midPoint) {\n    val tmp = this[index]\n    this[index] = this[reverseIndex]\n    this[reverseIndex] = tmp\n
reverseIndex--\n  }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n\npublic fun ByteArray.reverse():
Unit {\n  val midPoint = (size / 2) - 1\n  if (midPoint < 0) return\n  var reverseIndex = lastIndex\n  for (index in
0..midPoint) {\n    val tmp = this[index]\n    this[index] = this[reverseIndex]\n    this[reverseIndex] = tmp\n
reverseIndex--\n  }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n\npublic fun ShortArray.reverse():
Unit {\n  val midPoint = (size / 2) - 1\n  if (midPoint < 0) return\n  var reverseIndex = lastIndex\n  for (index in
0..midPoint) {\n    val tmp = this[index]\n    this[index] = this[reverseIndex]\n    this[reverseIndex] = tmp\n
reverseIndex--\n  }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n\npublic fun IntArray.reverse():
Unit {\n  val midPoint = (size / 2) - 1\n  if (midPoint < 0) return\n  var reverseIndex = lastIndex\n  for (index in
0..midPoint) {\n    val tmp = this[index]\n    this[index] = this[reverseIndex]\n    this[reverseIndex] = tmp\n
reverseIndex--\n  }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n\npublic fun LongArray.reverse():
Unit {\n  val midPoint = (size / 2) - 1\n  if (midPoint < 0) return\n  var reverseIndex = lastIndex\n  for (index in
0..midPoint) {\n    val tmp = this[index]\n    this[index] = this[reverseIndex]\n    this[reverseIndex] = tmp\n
reverseIndex--\n  }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n\npublic fun FloatArray.reverse():
Unit {\n  val midPoint = (size / 2) - 1\n  if (midPoint < 0) return\n  var reverseIndex = lastIndex\n  for (index in
0..midPoint) {\n    val tmp = this[index]\n    this[index] = this[reverseIndex]\n    this[reverseIndex] = tmp\n
reverseIndex--\n  }\n}\n\n/**\n * Reverses elements in the array in-place.\n */\n\npublic fun
DoubleArray.reverse(): Unit {\n  val midPoint = (size / 2) - 1\n  if (midPoint < 0) return\n  var reverseIndex =
lastIndex\n  for (index in 0..midPoint) {\n    val tmp = this[index]\n    this[index] = this[reverseIndex]\n
this[reverseIndex] = tmp\n    reverseIndex--\n  }\n}\n\n/**\n * Reverses elements in the array in-place.\n
*/\n\npublic fun BooleanArray.reverse(): Unit {\n  val midPoint = (size / 2) - 1\n  if (midPoint < 0) return\n  var
reverseIndex = lastIndex\n  for (index in 0..midPoint) {\n    val tmp = this[index]\n    this[index] =
this[reverseIndex]\n    this[reverseIndex] = tmp\n    reverseIndex--\n  }\n}\n\n/**\n * Reverses elements in the
array in-place.\n */\n\npublic fun CharArray.reverse(): Unit {\n  val midPoint = (size / 2) - 1\n  if (midPoint < 0)
return\n  var reverseIndex = lastIndex\n  for (index in 0..midPoint) {\n    val tmp = this[index]\n    this[index]
= this[reverseIndex]\n    this[reverseIndex] = tmp\n    reverseIndex--\n  }\n}\n\n/**\n * Returns a list with
elements in reversed order.\n */\n\npublic fun <T> Array<out T>.reversed(): List<T> {\n  if (isEmpty()) return
emptyList()\n  val list = toMutableList()\n  list.reverse()\n  return list\n}\n\n/**\n * Returns a list with elements
in reversed order.\n */\n\npublic fun ByteArray.reversed(): List<Byte> {\n  if (isEmpty()) return emptyList()\n  val
list = toMutableList()\n  list.reverse()\n  return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n
*/\n\npublic fun ShortArray.reversed(): List<Short> {\n  if (isEmpty()) return emptyList()\n  val list =
toMutableList()\n  list.reverse()\n  return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n
*/\n\npublic fun IntArray.reversed(): List<Int> {\n  if (isEmpty()) return emptyList()\n  val list = toMutableList()\n
list.reverse()\n  return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\n\npublic fun
LongArray.reversed(): List<Long> {\n  if (isEmpty()) return emptyList()\n  val list = toMutableList()\n  list.reverse()\n
return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\n\npublic fun
FloatArray.reversed(): List<Float> {\n  if (isEmpty()) return emptyList()\n  val list = toMutableList()\n  list.reverse()\n
return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\n\npublic fun
DoubleArray.reversed(): List<Double> {\n  if (isEmpty()) return emptyList()\n  val list = toMutableList()\n  list.reverse()\n
return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\n\npublic fun
BooleanArray.reversed(): List<Boolean> {\n  if (isEmpty()) return emptyList()\n  val list = toMutableList()\n

```

```

list.reverse()\n    return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun
CharArray.reversed(): List<Char> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n
list.reverse()\n    return list\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic
fun <T> Array<T>.reversedArray(): Array<T> {\n    if (isEmpty()) return this\n    val result = arrayOfNulls(this,
size)\n    val lastIndex = lastIndex\n    for (i in 0..lastIndex)\n        result[lastIndex - i] = this[i]\n    return
result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun
ByteArray.reversedArray(): ByteArray {\n    if (isEmpty()) return this\n    val result = ByteArray(size)\n    val
lastIndex = lastIndex\n    for (i in 0..lastIndex)\n        result[lastIndex - i] = this[i]\n    return result\n}\n\n/**\n *
Returns an array with elements of this array in reversed order.\n */\npublic fun ShortArray.reversedArray():
ShortArray {\n    if (isEmpty()) return this\n    val result = ShortArray(size)\n    val lastIndex = lastIndex\n    for (i in
0..lastIndex)\n        result[lastIndex - i] = this[i]\n    return result\n}\n\n/**\n * Returns an array with elements of this
array in reversed order.\n */\npublic fun IntArray.reversedArray(): IntArray {\n    if (isEmpty()) return this\n    val
result = IntArray(size)\n    val lastIndex = lastIndex\n    for (i in 0..lastIndex)\n        result[lastIndex - i] = this[i]\n
return result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun
LongArray.reversedArray(): LongArray {\n    if (isEmpty()) return this\n    val result = LongArray(size)\n    val
lastIndex = lastIndex\n    for (i in 0..lastIndex)\n        result[lastIndex - i] = this[i]\n    return result\n}\n\n/**\n *
Returns an array with elements of this array in reversed order.\n */\npublic fun FloatArray.reversedArray():
FloatArray {\n    if (isEmpty()) return this\n    val result = FloatArray(size)\n    val lastIndex = lastIndex\n    for (i in
0..lastIndex)\n        result[lastIndex - i] = this[i]\n    return result\n}\n\n/**\n * Returns an array with elements of this
array in reversed order.\n */\npublic fun DoubleArray.reversedArray(): DoubleArray {\n    if (isEmpty()) return
this\n    val result = DoubleArray(size)\n    val lastIndex = lastIndex\n    for (i in 0..lastIndex)\n        result[lastIndex -
i] = this[i]\n    return result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic
fun BooleanArray.reversedArray(): BooleanArray {\n    if (isEmpty()) return this\n    val result =
BooleanArray(size)\n    val lastIndex = lastIndex\n    for (i in 0..lastIndex)\n        result[lastIndex - i] = this[i]\n
return result\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n */\npublic fun
CharArray.reversedArray(): CharArray {\n    if (isEmpty()) return this\n    val result = CharArray(size)\n    val
lastIndex = lastIndex\n    for (i in 0..lastIndex)\n        result[lastIndex - i] = this[i]\n    return result\n}\n\n/**\n *
Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector]
function.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after
sorting.\n */\npublic inline fun <T, R : Comparable<R>> Array<out T>.sortBy(crossinline selector: (T) -> R?): Unit
{\n    if (size > 1) sortWith(compareBy(selector))\n}\n\n/**\n * Sorts elements in the array in-place descending
according to natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It
means that equal elements preserve their order relative to each other after sorting.\n */\npublic inline fun <T, R :
Comparable<R>> Array<out T>.sortByDescending(crossinline selector: (T) -> R?): Unit {\n    if (size > 1)
sortWith(compareByDescending(selector))\n}\n\n/**\n * Sorts elements in the array in-place descending according
to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to
each other after sorting.\n */\npublic fun <T : Comparable<T>> Array<out T>.sortDescending(): Unit {\n
sortWith(reverseOrder())\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural
sort order.\n */\npublic fun ByteArray.sortDescending(): Unit {\n    if (size > 1) {\n        sort()\n        reverse()\n
}\n}\n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n */\npublic fun
ShortArray.sortDescending(): Unit {\n    if (size > 1) {\n        sort()\n        reverse()\n    }\n}\n\n/**\n * Sorts
elements in the array in-place descending according to their natural sort order.\n */\npublic fun
IntArray.sortDescending(): Unit {\n    if (size > 1) {\n        sort()\n        reverse()\n    }\n}\n\n/**\n * Sorts elements
in the array in-place descending according to their natural sort order.\n */\npublic fun LongArray.sortDescending():
Unit {\n    if (size > 1) {\n        sort()\n        reverse()\n    }\n}\n\n/**\n * Sorts elements in the array in-place
descending according to their natural sort order.\n */\npublic fun FloatArray.sortDescending(): Unit {\n    if (size >
1) {\n        sort()\n        reverse()\n    }\n}\n\n/**\n * Sorts elements in the array in-place descending according
to their natural sort order.\n */\npublic fun DoubleArray.sortDescending(): Unit {\n    if (size > 1) {\n        sort()\n
}

```

```

reverse()\n } \n\n/**\n * Sorts elements in the array in-place descending according to their natural sort order.\n *\npublic fun CharArray.sortDescending(): Unit {\n if (size > 1) {\n sort()\n reverse()\n } \n} \n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n *\n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n *\npublic fun <T : Comparable<T>>\n Array<out T>.sorted(): List<T> {\n return sortedArray().asList()\n} \n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n *\npublic fun ByteArray.sorted(): List<Byte> {\n return\n toTypedArray().apply { sort() }.asList()\n} \n\n/**\n * Returns a list of all elements sorted according to their natural\n sort order.\n *\npublic fun ShortArray.sorted(): List<Short> {\n return toTypedArray().apply { sort()\n }.asList()\n} \n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n *\npublic fun\n IntArray.sorted(): List<Int> {\n return toTypedArray().apply { sort() }.asList()\n} \n\n/**\n * Returns a list of all\n elements sorted according to their natural sort order.\n *\npublic fun LongArray.sorted(): List<Long> {\n return\n toTypedArray().apply { sort() }.asList()\n} \n\n/**\n * Returns a list of all elements sorted according to their natural\n sort order.\n *\npublic fun FloatArray.sorted(): List<Float> {\n return toTypedArray().apply { sort()\n }.asList()\n} \n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n *\npublic fun\n DoubleArray.sorted(): List<Double> {\n return toTypedArray().apply { sort() }.asList()\n} \n\n/**\n * Returns a\n list of all elements sorted according to their natural sort order.\n *\npublic fun CharArray.sorted(): List<Char> {\n\n return toTypedArray().apply { sort() }.asList()\n} \n\n/**\n * Returns an array with all elements of this array sorted\n according to their natural sort order.\n *\n * The sort is _stable_. It means that equal elements preserve their order\n relative to each other after sorting.\n *\npublic fun <T : Comparable<T>> Array<T>.sortedArray(): Array<T> {\n\n if (isEmpty()) return this\n return this.copyOf().apply { sort() } \n} \n\n/**\n * Returns an array with all elements of\n this array sorted according to their natural sort order.\n *\npublic fun ByteArray.sortedArray(): ByteArray {\n\n if\n (isEmpty()) return this\n return this.copyOf().apply { sort() } \n} \n\n/**\n * Returns an array with all elements of\n this array sorted according to their natural sort order.\n *\npublic fun ShortArray.sortedArray(): ShortArray {\n\n if\n (isEmpty()) return this\n return this.copyOf().apply { sort() } \n} \n\n/**\n * Returns an array with all elements of\n this array sorted according to their natural sort order.\n *\npublic fun IntArray.sortedArray(): IntArray {\n\n if\n (isEmpty()) return this\n return this.copyOf().apply { sort() } \n} \n\n/**\n * Returns an array with all elements of\n this array sorted according to their natural sort order.\n *\npublic fun LongArray.sortedArray(): LongArray {\n\n if\n (isEmpty()) return this\n return this.copyOf().apply { sort() } \n} \n\n/**\n * Returns an array with all elements of\n this array sorted according to their natural sort order.\n *\npublic fun FloatArray.sortedArray(): FloatArray {\n\n if\n (isEmpty()) return this\n return this.copyOf().apply { sort() } \n} \n\n/**\n * Returns an array with all elements of\n this array sorted according to their natural sort order.\n *\npublic fun DoubleArray.sortedArray(): DoubleArray {\n\n if\n (isEmpty()) return this\n return this.copyOf().apply { sort() } \n} \n\n/**\n * Returns an array with all elements\n of this array sorted according to their natural sort order.\n *\npublic fun CharArray.sortedArray(): CharArray {\n\n if\n (isEmpty()) return this\n return this.copyOf().apply { sort() } \n} \n\n/**\n * Returns an array with all elements of\n this array sorted descending according to their natural sort order.\n *\n * The sort is _stable_. It means that equal\n elements preserve their order relative to each other after sorting.\n *\npublic fun <T : Comparable<T>>\n Array<T>.sortedArrayDescending(): Array<T> {\n\n if (isEmpty()) return this\n return this.copyOf().apply {\n sortWith(reverseOrder())\n } \n} \n\n/**\n * Returns an array with all elements of this array sorted descending\n according to their natural sort order.\n *\npublic fun ByteArray.sortedArrayDescending(): ByteArray {\n\n if\n (isEmpty()) return this\n return this.copyOf().apply { sortDescending() } \n} \n\n/**\n * Returns an array with all\n elements of this array sorted descending according to their natural sort order.\n *\npublic fun\n ShortArray.sortedArrayDescending(): ShortArray {\n\n if (isEmpty()) return this\n return this.copyOf().apply {\n sortDescending()\n } \n} \n\n/**\n * Returns an array with all elements of this array sorted descending according to\n their natural sort order.\n *\npublic fun IntArray.sortedArrayDescending(): IntArray {\n\n if (isEmpty()) return\n this\n return this.copyOf().apply { sortDescending() } \n} \n\n/**\n * Returns an array with all elements of this\n array sorted descending according to their natural sort order.\n *\npublic fun LongArray.sortedArrayDescending():\n LongArray {\n\n if (isEmpty()) return this\n return this.copyOf().apply { sortDescending() } \n} \n\n/**\n * Returns\n an array with all elements of this array sorted descending according to their natural sort order.\n *\npublic fun

```

```

FloatArray.sortedArrayDescending(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply {
        sortDescending()
    }
}
Returns an array with all elements of this array sorted descending according to their natural sort order.

public fun DoubleArray.sortedArrayDescending(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply {
        sortDescending()
    }
}
Returns an array with all elements of this array sorted descending according to their natural sort order.

public fun CharArray.sortedArrayDescending(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply {
        sortDescending()
    }
}
Returns an array with all elements of this array sorted according the specified [comparator].
The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

public fun <T> Array<out T>.sortedArrayWith(comparator: Comparator<in T>): Array<out T> {
    if (isEmpty()) return this
    return this.copyOf().apply {
        sortWith(comparator)
    }
}
Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

public inline fun <T, R : Comparable<R>> Array<out T>.sortedBy(crossinline selector: (T) -> R?): List<T> {
    return sortedWith(compareBy(selector))
}
Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> ByteArray.sortedBy(crossinline selector: (Byte) -> R?): List<Byte> {
    return sortedWith(compareBy(selector))
}
Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> ShortArray.sortedBy(crossinline selector: (Short) -> R?): List<Short> {
    return sortedWith(compareBy(selector))
}
Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> IntArray.sortedBy(crossinline selector: (Int) -> R?): List<Int> {
    return sortedWith(compareBy(selector))
}
Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> LongArray.sortedBy(crossinline selector: (Long) -> R?): List<Long> {
    return sortedWith(compareBy(selector))
}
Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> FloatArray.sortedBy(crossinline selector: (Float) -> R?): List<Float> {
    return sortedWith(compareBy(selector))
}
Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> DoubleArray.sortedBy(crossinline selector: (Double) -> R?): List<Double> {
    return sortedWith(compareBy(selector))
}
Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> BooleanArray.sortedBy(crossinline selector: (Boolean) -> R?): List<Boolean> {
    return sortedWith(compareBy(selector))
}
Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> CharArray.sortedBy(crossinline selector: (Char) -> R?): List<Char> {
    return sortedWith(compareBy(selector))
}
Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

public inline fun <T, R : Comparable<R>> Array<out T>.sortedByDescending(crossinline selector: (T) -> R?): List<T> {
    return sortedWith(compareByDescending(selector))
}
Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> ByteArray.sortedByDescending(crossinline selector: (Byte) -> R?): List<Byte> {
    return sortedWith(compareByDescending(selector))
}
Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> ShortArray.sortedByDescending(crossinline selector: (Short) -> R?): List<Short> {
    return sortedWith(compareByDescending(selector))
}
Returns a list of all elements sorted descending

```

according to natural sort order of the value returned by specified [selector] function.

```

public inline fun <R : Comparable<R>> IntArray.sortedByDescending(crossinline selector: (Int) -> R?): List<Int> {
    return sortedWith(compareByDescending(selector))
}

```

* Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

```

public inline fun <R : Comparable<R>> LongArray.sortedByDescending(crossinline selector: (Long) -> R?): List<Long> {
    return sortedWith(compareByDescending(selector))
}

```

* Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

```

public inline fun <R : Comparable<R>> FloatArray.sortedByDescending(crossinline selector: (Float) -> R?): List<Float> {
    return sortedWith(compareByDescending(selector))
}

```

* Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

```

public inline fun <R : Comparable<R>> DoubleArray.sortedByDescending(crossinline selector: (Double) -> R?): List<Double> {
    return sortedWith(compareByDescending(selector))
}

```

* Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

```

public inline fun <R : Comparable<R>> BooleanArray.sortedByDescending(crossinline selector: (Boolean) -> R?): List<Boolean> {
    return sortedWith(compareByDescending(selector))
}

```

* Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

```

public inline fun <R : Comparable<R>> CharArray.sortedByDescending(crossinline selector: (Char) -> R?): List<Char> {
    return sortedWith(compareByDescending(selector))
}

```

* Returns a list of all elements sorted descending according to their natural sort order.

* The sort is `_stable_`. It means that equal elements preserve their order relative to each other after sorting.

```

public fun <T : Comparable<T>> Array<out T>.sortedDescending(): List<T> {
    return sortedWith(reverseOrder())
}

```

* Returns a list of all elements sorted descending according to their natural sort order.

```

public fun ByteArray.sortedDescending(): List<Byte> {
    return copyOf().apply { sort() }.reversed()
}

```

* Returns a list of all elements sorted descending according to their natural sort order.

```

public fun ShortArray.sortedDescending(): List<Short> {
    return copyOf().apply { sort() }.reversed()
}

```

* Returns a list of all elements sorted descending according to their natural sort order.

```

public fun IntArray.sortedDescending(): List<Int> {
    return copyOf().apply { sort() }.reversed()
}

```

* Returns a list of all elements sorted descending according to their natural sort order.

```

public fun LongArray.sortedDescending(): List<Long> {
    return copyOf().apply { sort() }.reversed()
}

```

* Returns a list of all elements sorted descending according to their natural sort order.

```

public fun FloatArray.sortedDescending(): List<Float> {
    return copyOf().apply { sort() }.reversed()
}

```

* Returns a list of all elements sorted descending according to their natural sort order.

```

public fun DoubleArray.sortedDescending(): List<Double> {
    return copyOf().apply { sort() }.reversed()
}

```

* Returns a list of all elements sorted descending according to their natural sort order.

```

public fun CharArray.sortedDescending(): List<Char> {
    return copyOf().apply { sort() }.reversed()
}

```

* Returns a list of all elements sorted according to the specified [comparator].

* The sort is `_stable_`. It means that equal elements preserve their order relative to each other after sorting.

```

public fun <T> Array<out T>.sortedWith(comparator: Comparator<in T>): List<T> {
    return sortedArrayWith(comparator).asList()
}

```

* Returns a list of all elements sorted according to the specified [comparator].

```

public fun ByteArray.sortedWith(comparator: Comparator<in Byte>): List<Byte> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

```

* Returns a list of all elements sorted according to the specified [comparator].

```

public fun ShortArray.sortedWith(comparator: Comparator<in Short>): List<Short> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

```

* Returns a list of all elements sorted according to the specified [comparator].

```

public fun IntArray.sortedWith(comparator: Comparator<in Int>): List<Int> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

```

* Returns a list of all elements sorted according to the specified [comparator].

```

public fun LongArray.sortedWith(comparator: Comparator<in Long>): List<Long> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

```

* Returns a list of all elements sorted according to the specified [comparator].

```

public fun FloatArray.sortedWith(comparator: Comparator<in Float>): List<Float> {
    return

```

`toTypedArray().apply { sortBy(comparator) }.asList()` Returns a list of all elements sorted according to the specified [comparator].

`public fun DoubleArray.sortedWith(comparator: Comparator<in Double>): List<Double>` Returns a list of all elements sorted according to the specified [comparator].

`public fun BooleanArray.sortedWith(comparator: Comparator<in Boolean>): List<Boolean>` Returns a list of all elements sorted according to the specified [comparator].

`public fun CharArray.sortedWith(comparator: Comparator<in Char>): List<Char>` Returns a list of all elements sorted according to the specified [comparator].

`<T> Array<out T>.asList(): List<T>` Returns a [List] that wraps the original array.

`ByteArray.asList(): List<Byte>` Returns a [List] that wraps the original array.

`ShortArray.asList(): List<Short>` Returns a [List] that wraps the original array.

`IntArray.asList(): List<Int>` Returns a [List] that wraps the original array.

`LongArray.asList(): List<Long>` Returns a [List] that wraps the original array.

`FloatArray.asList(): List<Float>` Returns a [List] that wraps the original array.

`DoubleArray.asList(): List<Double>` Returns a [List] that wraps the original array.

`BooleanArray.asList(): List<Boolean>` Returns a [List] that wraps the original array.

`CharArray.asList(): List<Char>` Returns `true` if the two specified arrays are *deeply* equal to one another, i.e. contain the same number of the same elements in the same order. If two corresponding elements are nested arrays, they are also compared deeply. If any of arrays contains itself on any nesting level the behavior is undefined. The elements of other types are compared for equality with the `[Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

`@SinceKotlin("1.1") public expect infix fun <T> Array<out T>.contentDeepEquals(other: Array<out T>): Boolean` Returns a hash code based on the contents of this array as if it is [List]. Nested arrays are treated as lists too. If any of arrays contains itself on any nesting level the behavior is undefined.

`@SinceKotlin("1.1") public expect fun <T> Array<out T>.contentDeepHashCode(): Int` Returns a string representation of the contents of this array as if it is a [List]. Nested arrays are treated as lists too. If any of arrays contains itself on any nesting level that reference is rendered as `"[...]"` to prevent recursion.

`@sample samples.collections.Arrays.ContentOperations.contentDeepToString`

`@SinceKotlin("1.1") public expect fun <T> Array<out T>.contentDeepToString(): String` Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the `[Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

`@SinceKotlin("1.1") public expect infix fun <T> Array<out T>.contentEquals(other: Array<out T>): Boolean` Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the `[Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

`@SinceKotlin("1.1") public expect infix fun ByteArray.contentEquals(other: ByteArray): Boolean` Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the `[Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

`@SinceKotlin("1.1") public expect infix fun ShortArray.contentEquals(other: ShortArray): Boolean` Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the `[Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

`@SinceKotlin("1.1") public expect infix fun IntArray.contentEquals(other: IntArray): Boolean` Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the `[Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not

equal to `0.0`.
`@SinceKotlin("1.1")`
`public expect infix fun LongArray.contentEquals(other: LongArray): Boolean`
Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
`@SinceKotlin("1.1")`
`public expect infix fun FloatArray.contentEquals(other: FloatArray): Boolean`
Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
`@SinceKotlin("1.1")`
`public expect infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean`
Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
`@SinceKotlin("1.1")`
`public expect infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean`
Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the `[equals][Any.equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
`@SinceKotlin("1.1")`
`public expect infix fun CharArray.contentEquals(other: CharArray): Boolean`
Returns a hash code based on the contents of this array as if it is `[List]`.
`@SinceKotlin("1.1")`
`public expect fun <T> Array<out T>.contentHashCode(): Int`
Returns a hash code based on the contents of this array as if it is `[List]`.
`@SinceKotlin("1.1")`
`public expect fun ByteArray.contentHashCode(): Int`
Returns a hash code based on the contents of this array as if it is `[List]`.
`@SinceKotlin("1.1")`
`public expect fun ShortArray.contentHashCode(): Int`
Returns a hash code based on the contents of this array as if it is `[List]`.
`@SinceKotlin("1.1")`
`public expect fun IntArray.contentHashCode(): Int`
Returns a hash code based on the contents of this array as if it is `[List]`.
`@SinceKotlin("1.1")`
`public expect fun LongArray.contentHashCode(): Int`
Returns a hash code based on the contents of this array as if it is `[List]`.
`@SinceKotlin("1.1")`
`public expect fun FloatArray.contentHashCode(): Int`
Returns a hash code based on the contents of this array as if it is `[List]`.
`@SinceKotlin("1.1")`
`public expect fun DoubleArray.contentHashCode(): Int`
Returns a hash code based on the contents of this array as if it is `[List]`.
`@SinceKotlin("1.1")`
`public expect fun BooleanArray.contentHashCode(): Int`
Returns a hash code based on the contents of this array as if it is `[List]`.
`@SinceKotlin("1.1")`
`public expect fun CharArray.contentHashCode(): Int`
Returns a string representation of the contents of the specified array as if it is `[List]`.
`@sample` `samples.collections.Arrays.ContentOperations.contentToString`
`@SinceKotlin("1.1")`
`public expect fun <T> Array<out T>.contentToString(): String`
Returns a string representation of the contents of the specified array as if it is `[List]`.
`@sample` `samples.collections.Arrays.ContentOperations.contentToString`
`@SinceKotlin("1.1")`
`public expect fun ByteArray.contentToString(): String`
Returns a string representation of the contents of the specified array as if it is `[List]`.
`@sample` `samples.collections.Arrays.ContentOperations.contentToString`
`@SinceKotlin("1.1")`
`public expect fun ShortArray.contentToString(): String`
Returns a string representation of the contents of the specified array as if it is `[List]`.
`@sample` `samples.collections.Arrays.ContentOperations.contentToString`
`@SinceKotlin("1.1")`
`public expect fun IntArray.contentToString(): String`
Returns a string representation of the contents of the specified array as if it is `[List]`.
`@sample` `samples.collections.Arrays.ContentOperations.contentToString`
`@SinceKotlin("1.1")`
`public expect fun LongArray.contentToString(): String`
Returns a string representation of the contents of the specified array as if it is `[List]`.
`@sample` `samples.collections.Arrays.ContentOperations.contentToString`
`@SinceKotlin("1.1")`
`public expect fun FloatArray.contentToString(): String`
Returns a string representation of the contents of the specified array as if it is `[List]`.
`@sample` `samples.collections.Arrays.ContentOperations.contentToString`

*\n@SinceKotlin("1.1")\npublic expect fun DoubleArray.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin("1.1")\npublic expect fun BooleanArray.contentToString(): String\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n */\n@SinceKotlin("1.1")\npublic expect fun CharArray.contentToString(): String\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin("1.3")\npublic expect fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T>\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin("1.3")\npublic expect fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin("1.3")\npublic expect fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n@SinceKotlin("1.3")\npublic expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray\n\n/**\n * Copies this array or its subrange into the [destination]

array and returns that array.
 It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 @param destination the array to copy to.
 @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.
 @return the [destination] array.

```

@SinceKotlin("1.3")
public expect fun
LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): LongArray

```

Copies this array or its subrange into the [destination] array and returns that array.
 It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 @param destination the array to copy to.
 @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.
 @return the [destination] array.

```

@SinceKotlin("1.3")
public expect fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): FloatArray

```

Copies this array or its subrange into the [destination] array and returns that array.
 It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 @param destination the array to copy to.
 @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.
 @return the [destination] array.

```

@SinceKotlin("1.3")
public expect fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): DoubleArray

```

Copies this array or its subrange into the [destination] array and returns that array.
 It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 @param destination the array to copy to.
 @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.
 @return the [destination] array.

```

@SinceKotlin("1.3")
public expect fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): BooleanArray

```

Copies this array or its subrange into the [destination] array and returns that array.
 It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 @param destination the array to copy to.
 @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

> endIndex`.n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],.n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */\n\n@SinceKotlin("1.3")\npublic expect fun
CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
CharArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n */\n\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun <T> Array<T>.copyOf():
Array<T>\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n */\n\npublic expect fun ByteArray.copyOf():
ByteArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n */\n\npublic expect fun ShortArray.copyOf():
ShortArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n */\n\npublic expect fun IntArray.copyOf(): IntArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n */\n\npublic expect fun LongArray.copyOf():
LongArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n */\n\npublic expect fun FloatArray.copyOf():
FloatArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n */\n\npublic expect fun DoubleArray.copyOf():
DoubleArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n */\n\npublic expect fun BooleanArray.copyOf():
BooleanArray\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n */\n\npublic expect fun CharArray.copyOf():
CharArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * \n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n\n*\n */\n\npublic expect fun
ByteArray.copyOf(newSize: Int): ByteArray\n\n/**\n * Returns new array which is a copy of the original array,
resized to the given [newSize].\n * \n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n\n*\n */\n\npublic expect fun ShortArray.copyOf(newSize: Int): ShortArray\n\n/**\n * Returns new array which is a copy of the
original array, resized to the given [newSize].\n * \n * The copy is either truncated or padded at the end with zero values
if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the
[newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are
filled with zero values.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n\n*\n */\n\npublic expect fun IntArray.copyOf(newSize: Int): IntArray\n\n/**\n * Returns new array which is a copy of the
original array, resized to the given [newSize].\n * \n * The copy is either truncated or padded at the end with zero values
if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the
[newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are
filled with zero values.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n\n*\n */\n\npublic expect fun LongArray.copyOf(newSize: Int): LongArray\n\n/**\n * Returns new array which is a copy of
the original array, resized to the given [newSize].\n * \n * The copy is either truncated or padded at the end with zero
values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the
[newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are
filled with zero values.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n\n*\n */\n\npublic expect fun

*\npublic expect fun FloatArray.copyOf(newSize: Int): FloatArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun DoubleArray.copyOf(newSize: Int): DoubleArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with `false` values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `false` values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun BooleanArray.copyOf(newSize: Int): BooleanArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with null char (`\u0000`) values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with null char (`\u0000`) values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n */\npublic expect fun CharArray.copyOf(newSize: Int): CharArray\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with `null` values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `null` values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizingCopyOf\n */\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun <T> Array<T>.copyOf(newSize: Int): Array<T?>\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n */\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun <T> Array<T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T>\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n */\npublic expect fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n */\npublic expect fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n */\npublic expect fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n */\npublic expect fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n */\npublic expect fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n */\npublic expect fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the

range (exclusive), must be in `fromIndex..array.size`\n *\npublic expect fun
BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray\n\n/**\n * Returns a new array which is a
copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must
be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n
*\npublic expect fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray\n\n/**\n * Fills this array
or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive), 0 by
default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n * @throws
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n *\n@SinceKotlin("1.3")\npublic
expect fun <T> Array<T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or
its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive), 0 by
default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n * @throws
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n *\n@SinceKotlin("1.3")\npublic
expect fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or
its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive), 0 by
default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n * @throws
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n *\n@SinceKotlin("1.3")\npublic
expect fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or
its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive), 0 by
default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n * @throws
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n *\n@SinceKotlin("1.3")\npublic
expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or its
subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive), 0 by
default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n * @throws
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n *\n@SinceKotlin("1.3")\npublic
expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or
its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive), 0 by
default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n * @throws
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n *\n@SinceKotlin("1.3")\npublic
expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this array or
its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive), 0 by
default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n * @throws
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n *\n@SinceKotlin("1.3")\npublic
expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Fills this
array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range
(inclusive), 0 by default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n *
@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this
array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
*\n@SinceKotlin("1.3")\npublic expect fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int
= size): Unit\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param
fromIndex the start of the range (inclusive), 0 by default.\n * @param toIndex the end of the range (exclusive), size
of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex]

is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n *
 @SinceKotlin("1.3")\npublic expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit\n\n * Returns the range of valid indices for the array.\n *
 public val <T> Array<out T>.indices: IntRange\n get() = IntRange(0, lastIndex)\n\n * Returns the range of valid indices for the array.\n *
 public val ByteArray.indices: IntRange\n get() = IntRange(0, lastIndex)\n\n * Returns the range of valid indices for the array.\n *
 public val ShortArray.indices: IntRange\n get() = IntRange(0, lastIndex)\n\n * Returns the range of valid indices for the array.\n *
 public val IntArray.indices: IntRange\n get() = IntRange(0, lastIndex)\n\n * Returns the range of valid indices for the array.\n *
 public val LongArray.indices: IntRange\n get() = IntRange(0, lastIndex)\n\n * Returns the range of valid indices for the array.\n *
 public val FloatArray.indices: IntRange\n get() = IntRange(0, lastIndex)\n\n * Returns the range of valid indices for the array.\n *
 public val DoubleArray.indices: IntRange\n get() = IntRange(0, lastIndex)\n\n * Returns the range of valid indices for the array.\n *
 public val BooleanArray.indices: IntRange\n get() = IntRange(0, lastIndex)\n\n * Returns the range of valid indices for the array.\n *
 public val CharArray.indices: IntRange\n get() = IntRange(0, lastIndex)\n\n * Returns `true` if the array is empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.isEmpty(): Boolean {\n return size == 0\n}\n\n * Returns `true` if the array is empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun ByteArray.isEmpty(): Boolean {\n return size == 0\n}\n\n * Returns `true` if the array is empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun ShortArray.isEmpty(): Boolean {\n return size == 0\n}\n\n * Returns `true` if the array is empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun IntArray.isEmpty(): Boolean {\n return size == 0\n}\n\n * Returns `true` if the array is empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun LongArray.isEmpty(): Boolean {\n return size == 0\n}\n\n * Returns `true` if the array is empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun FloatArray.isEmpty(): Boolean {\n return size == 0\n}\n\n * Returns `true` if the array is empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun DoubleArray.isEmpty(): Boolean {\n return size == 0\n}\n\n * Returns `true` if the array is empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun BooleanArray.isEmpty(): Boolean {\n return size == 0\n}\n\n * Returns `true` if the array is empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun CharArray.isEmpty(): Boolean {\n return size == 0\n}\n\n * Returns `true` if the array is not empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.isNotEmpty(): Boolean {\n return !isEmpty()\n}\n\n * Returns `true` if the array is not empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun ByteArray.isNotEmpty(): Boolean {\n return !isEmpty()\n}\n\n * Returns `true` if the array is not empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun ShortArray.isNotEmpty(): Boolean {\n return !isEmpty()\n}\n\n * Returns `true` if the array is not empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun IntArray.isNotEmpty(): Boolean {\n return !isEmpty()\n}\n\n * Returns `true` if the array is not empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun LongArray.isNotEmpty(): Boolean {\n return !isEmpty()\n}\n\n * Returns `true` if the array is not empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun FloatArray.isNotEmpty(): Boolean {\n return !isEmpty()\n}\n\n * Returns `true` if the array is not empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun DoubleArray.isNotEmpty(): Boolean {\n return !isEmpty()\n}\n\n * Returns `true` if the array is not empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun BooleanArray.isNotEmpty(): Boolean {\n return !isEmpty()\n}\n\n * Returns `true` if the array is not empty.\n *
 @kotlin.internal.InlineOnly\npublic inline fun CharArray.isNotEmpty(): Boolean {\n return !isEmpty()\n}\n\n * Returns the last valid index for the array.\n *
 public val <T> Array<out T>.lastIndex: Int\n get() = size - 1\n\n * Returns the last valid index for the array.\n *
 public val ByteArray.lastIndex: Int\n get() = size - 1\n\n * Returns the last valid index for the array.\n *
 public val ShortArray.lastIndex: Int\n get() = size - 1\n\n * Returns the last valid index for the array.\n *
 public val IntArray.lastIndex: Int\n get() = size - 1\n\n * Returns the last valid index for the array.\n *
 public val LongArray.lastIndex: Int\n get() = size - 1\n\n * Returns the last valid index for the array.\n *
 public val FloatArray.lastIndex: Int\n get() = size - 1\n\n * Returns the last valid index for the array.\n *
 public val DoubleArray.lastIndex: Int\n get() = size - 1\n\n * Returns the last valid index for the

array.
`public val BooleanArray.lastIndex: Int` `get() = size - 1`
 * Returns the last valid index for the array.
`public val CharArray.lastIndex: Int` `get() = size - 1`
 * Returns an array containing all elements of the original array and then the given [element].
`@Suppress("NO_ACTUAL_FOR_EXPECT")` `public expect operator fun <T> Array<T>.plus(element: T): Array<T>`
 * Returns an array containing all elements of the original array and then the given [element].
`public expect operator fun ByteArray.plus(element: Byte): ByteArray`
 * Returns an array containing all elements of the original array and then the given [element].
`public expect operator fun ShortArray.plus(element: Short): ShortArray`
 * Returns an array containing all elements of the original array and then the given [element].
`public expect operator fun IntArray.plus(element: Int): IntArray`
 * Returns an array containing all elements of the original array and then the given [element].
`public expect operator fun LongArray.plus(element: Long): LongArray`
 * Returns an array containing all elements of the original array and then the given [element].
`public expect operator fun FloatArray.plus(element: Float): FloatArray`
 * Returns an array containing all elements of the original array and then the given [element].
`public expect operator fun DoubleArray.plus(element: Double): DoubleArray`
 * Returns an array containing all elements of the original array and then the given [element].
`public expect operator fun BooleanArray.plus(element: Boolean): BooleanArray`
 * Returns an array containing all elements of the original array and then the given [element].
`public expect operator fun CharArray.plus(element: Char): CharArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
`@Suppress("NO_ACTUAL_FOR_EXPECT")` `public expect operator fun <T> Array<T>.plus(elements: Collection<T>): Array<T>`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
`public expect operator fun ByteArray.plus(elements: Collection<Byte>): ByteArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
`public expect operator fun ShortArray.plus(elements: Collection<Short>): ShortArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
`public expect operator fun IntArray.plus(elements: Collection<Int>): IntArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
`public expect operator fun LongArray.plus(elements: Collection<Long>): LongArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
`public expect operator fun FloatArray.plus(elements: Collection<Float>): FloatArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
`public expect operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
`public expect operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
`public expect operator fun CharArray.plus(elements: Collection<Char>): CharArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
`@Suppress("NO_ACTUAL_FOR_EXPECT")` `public expect operator fun <T> Array<T>.plus(elements: Array<out T>): Array<T>`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
`public expect operator fun ByteArray.plus(elements: ByteArray): ByteArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
`public expect operator fun ShortArray.plus(elements: ShortArray): ShortArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
`public expect operator fun IntArray.plus(elements: IntArray): IntArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
`public expect operator fun LongArray.plus(elements: LongArray): LongArray`
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
`public expect operator fun FloatArray.plus(elements: FloatArray): FloatArray`
 * Returns an array containing all elements

of the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun
 DoubleArray.plus(elements: DoubleArray): DoubleArray\n\n/**\n * Returns an array containing all elements of the
 original array and then all elements of the given [elements] array.\n */\npublic expect operator fun
 BooleanArray.plus(elements: BooleanArray): BooleanArray\n\n/**\n * Returns an array containing all elements of
 the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun
 CharArray.plus(elements: CharArray): CharArray\n\n/**\n * Returns an array containing all elements of the original
 array and then the given [element].\n */\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun <T>
 Array<T>.plusElement(element: T): Array<T>\n\n/**\n * Sorts the array in-place.\n * \n * @sample
 samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun IntArray.sort(): Unit\n\n/**\n * Sorts the array
 in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun LongArray.sort():
 Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic
 expect fun ByteArray.sort(): Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample
 samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun ShortArray.sort(): Unit\n\n/**\n * Sorts the
 array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun
 DoubleArray.sort(): Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample
 samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun FloatArray.sort(): Unit\n\n/**\n * Sorts the
 array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun
 CharArray.sort(): Unit\n\n/**\n * Sorts the array in-place according to the natural order of its elements.\n * \n * The
 sort is `_stable_`. It means that equal elements preserve their order relative to each other after sorting.\n * \n *
 @sample samples.collections.Arrays.Sorting.sortArrayOfComparable\n */\npublic expect fun <T :
 Comparable<T>> Array<out T>.sort(): Unit\n\n/**\n * Sorts the array in-place according to the order specified by
 the given [comparator].\n * \n * The sort is `_stable_`. It means that equal elements preserve their order relative to
 each other after sorting.\n */\npublic expect fun <T> Array<out T>.sortWith(comparator: Comparator<in T>):
 Unit\n\n/**\n * Returns an array of Boolean containing all of the elements of this generic array.\n */\npublic fun
 Array<out Boolean>.toBooleanArray(): BooleanArray {\n return BooleanArray(size) { index -> this[index]
 }\n}\n\n/**\n * Returns an array of Byte containing all of the elements of this generic array.\n */\npublic fun
 Array<out Byte>.toByteArray(): ByteArray {\n return ByteArray(size) { index -> this[index] }\n}\n\n/**\n *
 Returns an array of Char containing all of the elements of this generic array.\n */\npublic fun Array<out
 Char>.toCharArray(): CharArray {\n return CharArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an
 array of Double containing all of the elements of this generic array.\n */\npublic fun Array<out
 Double>.toDoubleArray(): DoubleArray {\n return DoubleArray(size) { index -> this[index] }\n}\n\n/**\n *
 Returns an array of Float containing all of the elements of this generic array.\n */\npublic fun Array<out
 Float>.toFloatArray(): FloatArray {\n return FloatArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an
 array of Int containing all of the elements of this generic array.\n */\npublic fun Array<out Int>.toIntArray():
 IntArray {\n return IntArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of Long containing all of
 the elements of this generic array.\n */\npublic fun Array<out Long>.toLongArray(): LongArray {\n return
 LongArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of Short containing all of the elements of
 this generic array.\n */\npublic fun Array<out Short>.toShortArray(): ShortArray {\n return ShortArray(size) {\n
 index -> this[index] }\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive
 array.\n */\npublic expect fun ByteArray.toTypedArray(): Array<Byte>\n\n/**\n * Returns a *typed* object array
 containing all of the elements of this primitive array.\n */\npublic expect fun ShortArray.toTypedArray():
 Array<Short>\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */
 \npublic expect fun IntArray.toTypedArray(): Array<Int>\n\n/**\n * Returns a *typed* object array containing all
 of the elements of this primitive array.\n */\npublic expect fun LongArray.toTypedArray(): Array<Long>\n\n/**\n *
 Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic expect fun
 FloatArray.toTypedArray(): Array<Float>\n\n/**\n * Returns a *typed* object array containing all of the elements
 of this primitive array.\n */\npublic expect fun DoubleArray.toTypedArray(): Array<Double>\n\n/**\n * Returns a
typed object array containing all of the elements of this primitive array.\n */\npublic expect fun

BooleanArray.toArray(): Array<Boolean>\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n */\npublic expect fun CharArray.toArray(): Array<Char>\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <T, K, V> Array<out T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K, V> ByteArray.associate(transform: (Byte) -> Pair<K, V>): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K, V> ShortArray.associate(transform: (Short) -> Pair<K, V>): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K, V> IntArray.associate(transform: (Int) -> Pair<K, V>): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K, V> LongArray.associate(transform: (Long) -> Pair<K, V>): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K, V> FloatArray.associate(transform: (Float) -> Pair<K, V>): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K, V> DoubleArray.associate(transform: (Double) -> Pair<K, V>): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K, V> BooleanArray.associate(transform: (Boolean) -> Pair<K, V>): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by [transform] function\n * applied to elements of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K, V> CharArray.associate(transform: (Char) -> Pair<K, V>): Map<K, V> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n */

returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <T, K> Array<out T>.associateBy(keySelector: (T) -> K): Map<K, T> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K> ByteArray.associateBy(keySelector: (Byte) -> K): Map<K, Byte> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, Byte>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K> ShortArray.associateBy(keySelector: (Short) -> K): Map<K, Short> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, Short>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K> IntArray.associateBy(keySelector: (Int) -> K): Map<K, Int> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, Int>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K> LongArray.associateBy(keySelector: (Long) -> K): Map<K, Long> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, Long>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K> FloatArray.associateBy(keySelector: (Float) -> K): Map<K, Float> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, Float>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K> DoubleArray.associateBy(keySelector: (Double) -> K): Map<K, Double> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, Double>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K> BooleanArray.associateBy(keySelector: (Boolean) -> K): Map<K, Boolean> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, Boolean>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the elements from the given array indexed by the key\n * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n */\npublic inline fun <K> CharArray.associateBy(keySelector: (Char) -> K): Map<K, Char> {\n val capacity = mapCapacity(size).coerceAtLeast(16)\n return associateByTo(LinkedHashMap<K, Char>(capacity), keySelector)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform]

and indexed by [keySelector] functions applied to elements of the given array.

* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* The returned map preserves the entry iteration order of the original array.

```

public inline fun <T, K, V> Array<out T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

```

* Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.

* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* The returned map preserves the entry iteration order of the original array.

```

public inline fun <K, V> ByteArray.associateBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

```

* Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.

* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* The returned map preserves the entry iteration order of the original array.

```

public inline fun <K, V> ShortArray.associateBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

```

* Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.

* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* The returned map preserves the entry iteration order of the original array.

```

public inline fun <K, V> IntArray.associateBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

```

* Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.

* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* The returned map preserves the entry iteration order of the original array.

```

public inline fun <K, V> LongArray.associateBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

```

* Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.

* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* The returned map preserves the entry iteration order of the original array.

```

public inline fun <K, V> FloatArray.associateBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

```

* Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.

* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* The returned map preserves the entry iteration order of the original array.

```

public inline fun <K, V> DoubleArray.associateBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

```

* Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.

* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* The returned map preserves the entry iteration order of the original array.

```

public inline fun <K, V> BooleanArray.associateBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

```

* Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.

* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* The returned map preserves the entry iteration

```

order of the original array.\n */\npublic inline fun <K, V> CharArray.associateBy(keySelector: (Char) -> K,
valueTransform: (Char) -> V): Map<K, V> {\n    val capacity = mapCapacity(size).coerceAtLeast(16)\n    return
associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**\n * Populates and
returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function
applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would
have the same key returned by [keySelector] the last one gets added to the map.\n */\npublic inline fun <T, K, M :
MutableMap<in K, in T>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K): M {\n    for
(element in this) {\n        destination.put(keySelector(element), element)\n    }\n    return destination\n}\n\n/**\n *
Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the
[keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any
two elements would have the same key returned by [keySelector] the last one gets added to the map.\n */\npublic
inline fun <K, M : MutableMap<in K, in Byte>> ByteArray.associateByTo(destination: M, keySelector: (Byte) ->
K): M {\n    for (element in this) {\n        destination.put(keySelector(element), element)\n    }\n    return
destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is
provided by the [keySelector] function applied to each element of the given array\n * and value is the element
itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets
added to the map.\n */\npublic inline fun <K, M : MutableMap<in K, in Short>> ShortArray.associateByTo(destination: M,
keySelector: (Short) -> K): M {\n    for (element in this) {\n        destination.put(keySelector(element), element)\n
}\n    return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n *
where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the
element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets
added to the map.\n */\npublic inline fun <K, M : MutableMap<in K, in Int>> IntArray.associateByTo(destination:
M, keySelector: (Int) -> K): M {\n    for (element in this) {\n        destination.put(keySelector(element), element)\n
}\n    return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n *
where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the
element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets
added to the map.\n */\npublic inline fun <K, M : MutableMap<in K, in Long>>
LongArray.associateByTo(destination: M, keySelector: (Long) -> K): M {\n    for (element in this) {\n
destination.put(keySelector(element), element)\n    }\n    return destination\n}\n\n/**\n * Populates and returns the
[destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to
each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same
key returned by [keySelector] the last one gets added to the map.\n */\npublic inline fun <K, M : MutableMap<in K,
in Float>> FloatArray.associateByTo(destination: M, keySelector: (Float) -> K): M {\n    for (element in this) {\n
destination.put(keySelector(element), element)\n    }\n    return destination\n}\n\n/**\n * Populates and returns the
[destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to
each element of the given array\n * and value is the element itself.\n * \n * If any two elements would have the same
key returned by [keySelector] the last one gets added to the map.\n */\npublic inline fun <K, M : MutableMap<in K,
in Double>> DoubleArray.associateByTo(destination: M, keySelector: (Double) -> K): M {\n    for (element in this)
{\n        destination.put(keySelector(element), element)\n    }\n    return destination\n}\n\n/**\n * Populates and
returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function
applied to each element of the given array\n * and value is the element itself.\n * \n * If any two elements would
have the same key returned by [keySelector] the last one gets added to the map.\n */\npublic inline fun <K, M :
MutableMap<in K, in Boolean>> BooleanArray.associateByTo(destination: M, keySelector: (Boolean) -> K): M {\n
for (element in this) {\n        destination.put(keySelector(element), element)\n    }\n    return destination\n}\n\n/**
 * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the
[keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * If any
two elements would have the same key returned by [keySelector] the last one gets added to the map.\n */\npublic
inline fun <K, M : MutableMap<in K, in Char>> CharArray.associateByTo(destination: M, keySelector: (Char) ->

```

K): M { \n for (element in this) { \n destination.put(keySelector(element), element)\n } \n return destination\n} \n\n/** \n * Populates and returns the [destination] mutable map with key-value pairs, \n * where key is provided by the [keySelector] function and \n * and value is provided by the [valueTransform] function applied to elements of the given array. \n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. \n */ \n\npublic inline fun <T, K, V, M : MutableMap<in K, in V>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M { \n for (element in this) { \n destination.put(keySelector(element), valueTransform(element))\n } \n return destination\n} \n\n/** \n * Populates and returns the [destination] mutable map with key-value pairs, \n * where key is provided by the [keySelector] function and \n * and value is provided by the [valueTransform] function applied to elements of the given array. \n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. \n */ \n\npublic inline fun <K, V, M : MutableMap<in K, in V>> ByteArray.associateByTo(destination: M, keySelector: (Byte) -> K, valueTransform: (Byte) -> V): M { \n for (element in this) { \n destination.put(keySelector(element), valueTransform(element))\n } \n return destination\n} \n\n/** \n * Populates and returns the [destination] mutable map with key-value pairs, \n * where key is provided by the [keySelector] function and \n * and value is provided by the [valueTransform] function applied to elements of the given array. \n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. \n */ \n\npublic inline fun <K, V, M : MutableMap<in K, in V>> ShortArray.associateByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M { \n for (element in this) { \n destination.put(keySelector(element), valueTransform(element))\n } \n return destination\n} \n\n/** \n * Populates and returns the [destination] mutable map with key-value pairs, \n * where key is provided by the [keySelector] function and \n * and value is provided by the [valueTransform] function applied to elements of the given array. \n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. \n */ \n\npublic inline fun <K, V, M : MutableMap<in K, in V>> IntArray.associateByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M { \n for (element in this) { \n destination.put(keySelector(element), valueTransform(element))\n } \n return destination\n} \n\n/** \n * Populates and returns the [destination] mutable map with key-value pairs, \n * where key is provided by the [keySelector] function and \n * and value is provided by the [valueTransform] function applied to elements of the given array. \n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. \n */ \n\npublic inline fun <K, V, M : MutableMap<in K, in V>> LongArray.associateByTo(destination: M, keySelector: (Long) -> K, valueTransform: (Long) -> V): M { \n for (element in this) { \n destination.put(keySelector(element), valueTransform(element))\n } \n return destination\n} \n\n/** \n * Populates and returns the [destination] mutable map with key-value pairs, \n * where key is provided by the [keySelector] function and \n * and value is provided by the [valueTransform] function applied to elements of the given array. \n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. \n */ \n\npublic inline fun <K, V, M : MutableMap<in K, in V>> FloatArray.associateByTo(destination: M, keySelector: (Float) -> K, valueTransform: (Float) -> V): M { \n for (element in this) { \n destination.put(keySelector(element), valueTransform(element))\n } \n return destination\n} \n\n/** \n * Populates and returns the [destination] mutable map with key-value pairs, \n * where key is provided by the [keySelector] function and \n * and value is provided by the [valueTransform] function applied to elements of the given array. \n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. \n */ \n\npublic inline fun <K, V, M : MutableMap<in K, in V>> DoubleArray.associateByTo(destination: M, keySelector: (Double) -> K, valueTransform: (Double) -> V): M { \n for (element in this) { \n destination.put(keySelector(element), valueTransform(element))\n } \n return destination\n} \n\n/** \n * Populates and returns the [destination] mutable map with key-value pairs, \n * where key is provided by the [keySelector] function and \n * and value is provided by the [valueTransform] function applied to elements of the given array. \n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. \n */ \n\npublic inline fun <K, V, M : MutableMap<in K, in V>> BooleanArray.associateByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M { \n for (element in this) { \n destination.put(keySelector(element),

```

valueTransform(element))\n } return destination\n\n\n * Populates and returns the [destination] mutable
map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by
the [valueTransform] function applied to elements of the given array.\n * \n * If any two elements would have the
same key returned by [keySelector] the last one gets added to the map.\n */\npublic inline fun <K, V, M :
MutableMap<in K, in V>> CharArray.associateByTo(destination: M, keySelector: (Char) -> K, valueTransform:
(Char) -> V): M {\n for (element in this) {\n destination.put(keySelector(element),
valueTransform(element))\n }\n return destination\n\n\n * Populates and returns the [destination] mutable
map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n *
If any of two pairs would have the same key the last one gets added to the map.\n */\npublic inline fun <T, K, V, M :
MutableMap<in K, in V>> Array<out T>.associateTo(destination: M, transform: (T) -> Pair<K, V>): M {\n for
(element in this) {\n destination += transform(element)\n }\n return destination\n\n\n * Populates and
returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each
element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the
map.\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> ByteArray.associateTo(destination: M,
transform: (Byte) -> Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n
return destination\n\n\n * Populates and returns the [destination] mutable map with key-value pairs\n *
provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have
the same key the last one gets added to the map.\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>>
ShortArray.associateTo(destination: M, transform: (Short) -> Pair<K, V>): M {\n for (element in this) {\n
destination += transform(element)\n }\n return destination\n\n\n * Populates and returns the [destination]
mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n
* \n * If any of two pairs would have the same key the last one gets added to the map.\n */\npublic inline fun <K, V,
M : MutableMap<in K, in V>> IntArray.associateTo(destination: M, transform: (Int) -> Pair<K, V>): M {\n for
(element in this) {\n destination += transform(element)\n }\n return destination\n\n\n * Populates and
returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each
element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the
map.\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>> LongArray.associateTo(destination: M,
transform: (Long) -> Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n
return destination\n\n\n * Populates and returns the [destination] mutable map with key-value pairs\n *
provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have
the same key the last one gets added to the map.\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>>
FloatArray.associateTo(destination: M, transform: (Float) -> Pair<K, V>): M {\n for (element in this) {\n
destination += transform(element)\n }\n return destination\n\n\n * Populates and returns the [destination]
mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n
* \n * If any of two pairs would have the same key the last one gets added to the map.\n */\npublic inline fun <K, V,
M : MutableMap<in K, in V>> DoubleArray.associateTo(destination: M, transform: (Double) -> Pair<K, V>): M
{\n for (element in this) {\n destination += transform(element)\n }\n return destination\n\n\n *
Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function
applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets
added to the map.\n */\npublic inline fun <K, V, M : MutableMap<in K, in V>>
BooleanArray.associateTo(destination: M, transform: (Boolean) -> Pair<K, V>): M {\n for (element in this) {\n
destination += transform(element)\n }\n return destination\n\n\n * Populates and returns the [destination]
mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n
* \n * If any of two pairs would have the same key the last one gets added to the map.\n */\npublic inline fun <K, V,
M : MutableMap<in K, in V>> CharArray.associateTo(destination: M, transform: (Char) -> Pair<K, V>): M {\n
for (element in this) {\n destination += transform(element)\n }\n return destination\n\n\n * Appends
all elements to the given [destination] collection.\n */\npublic fun <T, C : MutableCollection<in T>> Array<out
T>.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return

```

```

destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n *\npublic fun <C :
MutableCollection<in Byte>> ByteArray.toCollection(destination: C): C {\n  for (item in this) {\n
destination.add(item)\n  }\n  return destination\n}\n\n/**\n * Appends all elements to the given [destination]
collection.\n *\npublic fun <C : MutableCollection<in Short>> ShortArray.toCollection(destination: C): C {\n  for
(item in this) {\n    destination.add(item)\n  }\n  return destination\n}\n\n/**\n * Appends all elements to the
given [destination] collection.\n *\npublic fun <C : MutableCollection<in Int>> IntArray.toCollection(destination:
C): C {\n  for (item in this) {\n    destination.add(item)\n  }\n  return destination\n}\n\n/**\n * Appends all
elements to the given [destination] collection.\n *\npublic fun <C : MutableCollection<in Long>>
LongArray.toCollection(destination: C): C {\n  for (item in this) {\n    destination.add(item)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n *\npublic fun <C :
MutableCollection<in Float>> FloatArray.toCollection(destination: C): C {\n  for (item in this) {\n
destination.add(item)\n  }\n  return destination\n}\n\n/**\n * Appends all elements to the given [destination]
collection.\n *\npublic fun <C : MutableCollection<in Double>> DoubleArray.toCollection(destination: C): C {\n
for (item in this) {\n    destination.add(item)\n  }\n  return destination\n}\n\n/**\n * Appends all elements to the
given [destination] collection.\n *\npublic fun <C : MutableCollection<in Boolean>>
BooleanArray.toCollection(destination: C): C {\n  for (item in this) {\n    destination.add(item)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n *\npublic fun <C :
MutableCollection<in Char>> CharArray.toCollection(destination: C): C {\n  for (item in this) {\n
destination.add(item)\n  }\n  return destination\n}\n\n/**\n * Returns a [HashSet] of all elements.\n *\npublic fun
<T> Array<out T>.toHashSet(): HashSet<T> {\n  return
toCollection(HashSet<T>(mapCapacity(size)))\n}\n\n/**\n * Returns a [HashSet] of all elements.\n *\npublic fun
ByteArray.toHashSet(): HashSet<Byte> {\n  return toCollection(HashSet<Byte>(mapCapacity(size)))\n}\n\n/**\n
* Returns a [HashSet] of all elements.\n *\npublic fun ShortArray.toHashSet(): HashSet<Short> {\n  return
toCollection(HashSet<Short>(mapCapacity(size)))\n}\n\n/**\n * Returns a [HashSet] of all elements.\n *\npublic
fun IntArray.toHashSet(): HashSet<Int> {\n  return toCollection(HashSet<Int>(mapCapacity(size)))\n}\n\n/**\n
* Returns a [HashSet] of all elements.\n *\npublic fun LongArray.toHashSet(): HashSet<Long> {\n  return
toCollection(HashSet<Long>(mapCapacity(size)))\n}\n\n/**\n * Returns a [HashSet] of all elements.\n *\npublic
fun FloatArray.toHashSet(): HashSet<Float> {\n  return
toCollection(HashSet<Float>(mapCapacity(size)))\n}\n\n/**\n * Returns a [HashSet] of all elements.\n *\npublic
fun DoubleArray.toHashSet(): HashSet<Double> {\n  return
toCollection(HashSet<Double>(mapCapacity(size)))\n}\n\n/**\n * Returns a [HashSet] of all elements.\n *\npublic
fun BooleanArray.toHashSet(): HashSet<Boolean> {\n  return
toCollection(HashSet<Boolean>(mapCapacity(size)))\n}\n\n/**\n * Returns a [HashSet] of all elements.\n
*\npublic fun CharArray.toHashSet(): HashSet<Char> {\n  return
toCollection(HashSet<Char>(mapCapacity(size)))\n}\n\n/**\n * Returns a [List] containing all elements.\n
*\npublic fun <T> Array<out T>.toList(): List<T> {\n  return when (size) {\n    0 -> emptyList()\n    1 ->
listOf(this[0])\n    else -> this.toMutableList()\n  }\n}\n\n/**\n * Returns a [List] containing all elements.\n
*\npublic fun ByteArray.toList(): List<Byte> {\n  return when (size) {\n    0 -> emptyList()\n    1 ->
listOf(this[0])\n    else -> this.toMutableList()\n  }\n}\n\n/**\n * Returns a [List] containing all elements.\n
*\npublic fun ShortArray.toList(): List<Short> {\n  return when (size) {\n    0 -> emptyList()\n    1 ->
listOf(this[0])\n    else -> this.toMutableList()\n  }\n}\n\n/**\n * Returns a [List] containing all elements.\n
*\npublic fun IntArray.toList(): List<Int> {\n  return when (size) {\n    0 -> emptyList()\n    1 ->
listOf(this[0])\n    else -> this.toMutableList()\n  }\n}\n\n/**\n * Returns a [List] containing all elements.\n
*\npublic fun LongArray.toList(): List<Long> {\n  return when (size) {\n    0 -> emptyList()\n    1 ->
listOf(this[0])\n    else -> this.toMutableList()\n  }\n}\n\n/**\n * Returns a [List] containing all elements.\n
*\npublic fun FloatArray.toList(): List<Float> {\n  return when (size) {\n    0 -> emptyList()\n    1 ->
listOf(this[0])\n    else -> this.toMutableList()\n  }\n}\n\n/**\n * Returns a [List] containing all elements.\n
*\npublic fun DoubleArray.toList(): List<Double> {\n  return when (size) {\n    0 -> emptyList()\n    1 ->

```

```

listOf(this[0])\n     else -> this.toMutableList()\n }]\n\n/**\n * Returns a [List] containing all elements.\n *\npublic fun BooleanArray.toList(): List<Boolean> {\n return when (size) {\n 0 -> emptyList()\n 1 ->
listOf(this[0])\n     else -> this.toMutableList()\n }]\n\n/**\n * Returns a [List] containing all elements.\n *\npublic fun CharArray.toList(): List<Char> {\n return when (size) {\n 0 -> emptyList()\n 1 ->
listOf(this[0])\n     else -> this.toMutableList()\n }]\n\n/**\n * Returns a [MutableList] filled with all elements
of this array.\n *\npublic fun <T> Array<out T>.toMutableList(): MutableList<T> {\n return
ArrayList(this.asCollection())}\n\n/**\n * Returns a [MutableList] filled with all elements of this array.\n *\npublic fun ByteArray.toMutableList(): MutableList<Byte> {\n val list = ArrayList<Byte>(size)\n for (item
in this) list.add(item)\n return list}\n\n/**\n * Returns a [MutableList] filled with all elements of this array.\n *\npublic fun ShortArray.toMutableList(): MutableList<Short> {\n val list = ArrayList<Short>(size)\n for (item
in this) list.add(item)\n return list}\n\n/**\n * Returns a [MutableList] filled with all elements of this array.\n *\npublic fun IntArray.toMutableList(): MutableList<Int> {\n val list = ArrayList<Int>(size)\n for (item in this)
list.add(item)\n return list}\n\n/**\n * Returns a [MutableList] filled with all elements of this array.\n *\npublic fun LongArray.toMutableList(): MutableList<Long> {\n val list = ArrayList<Long>(size)\n for (item in this)
list.add(item)\n return list}\n\n/**\n * Returns a [MutableList] filled with all elements of this array.\n *\npublic fun FloatArray.toMutableList(): MutableList<Float> {\n val list = ArrayList<Float>(size)\n for (item in this)
list.add(item)\n return list}\n\n/**\n * Returns a [MutableList] filled with all elements of this array.\n *\npublic fun DoubleArray.toMutableList(): MutableList<Double> {\n val list = ArrayList<Double>(size)\n for (item in
this) list.add(item)\n return list}\n\n/**\n * Returns a [MutableList] filled with all elements of this array.\n *\npublic fun BooleanArray.toMutableList(): MutableList<Boolean> {\n val list = ArrayList<Boolean>(size)\n
for (item in this) list.add(item)\n return list}\n\n/**\n * Returns a [MutableList] filled with all elements of this
array.\n *\npublic fun CharArray.toMutableList(): MutableList<Char> {\n val list = ArrayList<Char>(size)\n
for (item in this) list.add(item)\n return list}\n\n/**\n * Returns a [Set] of all elements.\n *\n * The returned set
preserves the element iteration order of the original array.\n *\npublic fun <T> Array<out T>.toSet(): Set<T> {\n
return when (size) {\n 0 -> emptySet()\n 1 -> setOf(this[0])\n else ->
toCollection(LinkedHashSet<T>(mapCapacity(size)))}\n }]\n\n/**\n * Returns a [Set] of all elements.\n *\n\n *
The returned set preserves the element iteration order of the original array.\n *\npublic fun ByteArray.toSet():
Set<Byte> {\n return when (size) {\n 0 -> emptySet()\n 1 -> setOf(this[0])\n else ->
toCollection(LinkedHashSet<Byte>(mapCapacity(size)))}\n }]\n\n/**\n * Returns a [Set] of all elements.\n *\n\n *
The returned set preserves the element iteration order of the original array.\n *\npublic fun ShortArray.toSet():
Set<Short> {\n return when (size) {\n 0 -> emptySet()\n 1 -> setOf(this[0])\n else ->
toCollection(LinkedHashSet<Short>(mapCapacity(size)))}\n }]\n\n/**\n * Returns a [Set] of all elements.\n *\n\n *
The returned set preserves the element iteration order of the original array.\n *\npublic fun IntArray.toSet():
Set<Int> {\n return when (size) {\n 0 -> emptySet()\n 1 -> setOf(this[0])\n else ->
toCollection(LinkedHashSet<Int>(mapCapacity(size)))}\n }]\n\n/**\n * Returns a [Set] of all elements.\n *\n\n *
The returned set preserves the element iteration order of the original array.\n *\npublic fun LongArray.toSet():
Set<Long> {\n return when (size) {\n 0 -> emptySet()\n 1 -> setOf(this[0])\n else ->
toCollection(LinkedHashSet<Long>(mapCapacity(size)))}\n }]\n\n/**\n * Returns a [Set] of all elements.\n *\n\n *
The returned set preserves the element iteration order of the original array.\n *\npublic fun FloatArray.toSet():
Set<Float> {\n return when (size) {\n 0 -> emptySet()\n 1 -> setOf(this[0])\n else ->
toCollection(LinkedHashSet<Float>(mapCapacity(size)))}\n }]\n\n/**\n * Returns a [Set] of all elements.\n *\n\n *
The returned set preserves the element iteration order of the original array.\n *\npublic fun DoubleArray.toSet():
Set<Double> {\n return when (size) {\n 0 -> emptySet()\n 1 -> setOf(this[0])\n else ->
toCollection(LinkedHashSet<Double>(mapCapacity(size)))}\n }]\n\n/**\n * Returns a [Set] of all elements.\n *\n\n *
The returned set preserves the element iteration order of the original array.\n *\npublic fun
BooleanArray.toSet(): Set<Boolean> {\n return when (size) {\n 0 -> emptySet()\n 1 -> setOf(this[0])\n
else -> toCollection(LinkedHashSet<Boolean>(mapCapacity(size)))}\n }]\n\n/**\n * Returns a [Set] of all
elements.\n *\n\n * The returned set preserves the element iteration order of the original array.\n *\npublic fun

```



```

CharArray.toSet(): Set<Char> {\n  return when (size) {\n    0 -> emptySet()\n    1 -> setOf(this[0])\n    else -
> toCollection(LinkedHashSet<Char>(mapCapacity(size)))\n  }\n}\n\n/**\n * Returns a single list of all elements
yielded from results of [transform] function being invoked on each element of original array.\n */\npublic inline fun
<T, R> Array<out T>.flatMap(transform: (T) -> Iterable<R>): List<R> {\n  return flatMapTo(ArrayList<R>(),
transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being
invoked on each element of original array.\n */\npublic inline fun <R> ByteArray.flatMap(transform: (Byte) ->
Iterable<R>): List<R> {\n  return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all
elements yielded from results of [transform] function being invoked on each element of original array.\n */\npublic
inline fun <R> ShortArray.flatMap(transform: (Short) -> Iterable<R>): List<R> {\n  return
flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of
[transform] function being invoked on each element of original array.\n */\npublic inline fun <R>
IntArray.flatMap(transform: (Int) -> Iterable<R>): List<R> {\n  return flatMapTo(ArrayList<R>(),
transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being
invoked on each element of original array.\n */\npublic inline fun <R> LongArray.flatMap(transform: (Long) ->
Iterable<R>): List<R> {\n  return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all
elements yielded from results of [transform] function being invoked on each element of original array.\n */\npublic
inline fun <R> FloatArray.flatMap(transform: (Float) -> Iterable<R>): List<R> {\n  return
flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of
[transform] function being invoked on each element of original array.\n */\npublic inline fun <R>
DoubleArray.flatMap(transform: (Double) -> Iterable<R>): List<R> {\n  return flatMapTo(ArrayList<R>(),
transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being
invoked on each element of original array.\n */\npublic inline fun <R> BooleanArray.flatMap(transform: (Boolean)
-> Iterable<R>): List<R> {\n  return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Returns a single list of
all elements yielded from results of [transform] function being invoked on each element of original array.\n */\n
*\npublic inline fun <R> CharArray.flatMap(transform: (Char) -> Iterable<R>): List<R> {\n  return
flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Appends all elements yielded from results of [transform]
function being invoked on each element of original array, to the given [destination].\n */\npublic inline fun <T, R, C
: MutableCollection<in R>> Array<out T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {\n  for
(element in this) {\n    val list = transform(element)\n    destination.addAll(list)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each
element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>>
ByteArray.flatMapTo(destination: C, transform: (Byte) -> Iterable<R>): C {\n  for (element in this) {\n    val list
= transform(element)\n    destination.addAll(list)\n  }\n  return destination\n}\n\n/**\n * Appends all elements
yielded from results of [transform] function being invoked on each element of original array, to the given
[destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> ShortArray.flatMapTo(destination: C,
transform: (Short) -> Iterable<R>): C {\n  for (element in this) {\n    val list = transform(element)\n
destination.addAll(list)\n  }\n  return destination\n}\n\n/**\n * Appends all elements yielded from results of
[transform] function being invoked on each element of original array, to the given [destination].\n */\npublic inline
fun <R, C : MutableCollection<in R>> IntArray.flatMapTo(destination: C, transform: (Int) -> Iterable<R>): C {\n
for (element in this) {\n    val list = transform(element)\n    destination.addAll(list)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each
element of original array, to the given [destination].\n */\npublic inline fun <R, C : MutableCollection<in R>>
LongArray.flatMapTo(destination: C, transform: (Long) -> Iterable<R>): C {\n  for (element in this) {\n    val
list = transform(element)\n    destination.addAll(list)\n  }\n  return destination\n}\n\n/**\n * Appends all
elements yielded from results of [transform] function being invoked on each element of original array, to the given
[destination].\n */\npublic inline fun <R, C : MutableCollection<in R>> FloatArray.flatMapTo(destination: C,
transform: (Float) -> Iterable<R>): C {\n  for (element in this) {\n    val list = transform(element)\n
destination.addAll(list)\n  }\n  return destination\n}\n\n/**\n * Appends all elements yielded from results of

```

```

[transform] function being invoked on each element of original array, to the given [destination].\n *\npublic inline
fun <R, C : MutableCollection<in R>> DoubleArray.flatMapTo(destination: C, transform: (Double) ->
Iterable<R>): C {\n    for (element in this) {\n        val list = transform(element)\n        destination.addAll(list)\n    }\n    return destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being
invoked on each element of original array, to the given [destination].\n *\npublic inline fun <R, C :
MutableCollection<in R>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable<R>): C {\n
for (element in this) {\n    val list = transform(element)\n    destination.addAll(list)\n } \n return
destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each
element of original array, to the given [destination].\n *\npublic inline fun <R, C : MutableCollection<in R>>
CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {\n    for (element in this) {\n        val list
= transform(element)\n        destination.addAll(list)\n    }\n    return destination\n}\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map
where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the
entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupBy\n *\npublic inline fun <T, K> Array<out
T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {\n    return groupByTo(LinkedHashMap<K,
MutableList<T>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the
given [keySelector] function\n * applied to each element and returns a map where each group key is associated with
a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced
from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n *\npublic
inline fun <K> ByteArray.groupBy(keySelector: (Byte) -> K): Map<K, List<Byte>> {\n    return
groupByTo(LinkedHashMap<K, MutableList<Byte>>(), keySelector)\n}\n\n/**\n * Groups elements of the original
array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where
each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the entry
iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupBy\n *\npublic inline fun <K>
ShortArray.groupBy(keySelector: (Short) -> K): Map<K, List<Short>> {\n    return
groupByTo(LinkedHashMap<K, MutableList<Short>>(), keySelector)\n}\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map
where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the
entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupBy\n *\npublic inline fun <K>
IntArray.groupBy(keySelector: (Int) -> K): Map<K, List<Int>> {\n    return groupByTo(LinkedHashMap<K,
MutableList<Int>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the
given [keySelector] function\n * applied to each element and returns a map where each group key is associated with
a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced
from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n *\npublic
inline fun <K> LongArray.groupBy(keySelector: (Long) -> K): Map<K, List<Long>> {\n    return
groupByTo(LinkedHashMap<K, MutableList<Long>>(), keySelector)\n}\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map
where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the
entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupBy\n *\npublic inline fun <K>
FloatArray.groupBy(keySelector: (Float) -> K): Map<K, List<Float>> {\n    return groupByTo(LinkedHashMap<K,
MutableList<Float>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the
given [keySelector] function\n * applied to each element and returns a map where each group key is associated with
a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced
from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n *\npublic

```

```

inline fun <K> DoubleArray.groupBy(keySelector: (Double) -> K): Map<K, List<Double>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<Double>>(), keySelector)\n}\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map
where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the
entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K>
BooleanArray.groupBy(keySelector: (Boolean) -> K): Map<K, List<Boolean>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<Boolean>>(), keySelector)\n}\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map
where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the
entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K>
CharArray.groupBy(keySelector: (Char) -> K): Map<K, List<Char>> {\n  return groupByTo(LinkedHashMap<K,
MutableList<Char>>(), keySelector)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied
to each element of the original array\n * by the key returned by the given [keySelector] function applied to the
element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The
returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <T, K, V>
Array<out T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values
returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by
the given [keySelector] function applied to the element\n * and returns a map where each group key is associated
with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V>
ByteArray.groupBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values
returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by
the given [keySelector] function applied to the element\n * and returns a map where each group key is associated
with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V>
ShortArray.groupBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values
returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by
the given [keySelector] function applied to the element\n * and returns a map where each group key is associated
with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V>
IntArray.groupBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values
returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by
the given [keySelector] function applied to the element\n * and returns a map where each group key is associated
with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V>
LongArray.groupBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values

```

returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample

```

samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic inline fun <K, V>
FloatArray.groupBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n }\n\n/**\n * Groups values
returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic inline fun <K, V>
DoubleArray.groupBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, List<V>> {\n
return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n }\n\n/**\n * Groups
values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic inline fun <K, V>
BooleanArray.groupBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, List<V>> {\n
return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n }\n\n/**\n * Groups
values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic inline fun <K, V>
CharArray.groupBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n }\n\n/**\n * Groups elements
of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline
fun <T, K, M : MutableMap<in K, MutableList<T>>> Array<out T>.groupByTo(destination: M, keySelector: (T) -
> K): M {\n  for (element in this) {\n    val key = keySelector(element)\n    val list = destination.getOrPut(key)
{ ArrayList<T>() }\n    list.add(element)\n  }\n  return destination\n }\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline
fun <K, M : MutableMap<in K, MutableList<Byte>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -
> K): M {\n  for (element in this) {\n    val key = keySelector(element)\n    val list = destination.getOrPut(key)
{ ArrayList<Byte>() }\n    list.add(element)\n  }\n  return destination\n }\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline
fun <K, M : MutableMap<in K, MutableList<Short>>> ShortArray.groupByTo(destination: M, keySelector: (Short)
-> K): M {\n  for (element in this) {\n    val key = keySelector(element)\n    val list =
destination.getOrPut(key) { ArrayList<Short>() }\n    list.add(element)\n  }\n  return destination\n }\n\n/**\n *
Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n *

```

```

@return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*/\npublic inline fun <K, M : MutableMap<in K, MutableList<Int>>> IntArray.groupByTo(destination: M,\n
keySelector: (Int) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list =\n
destination.getOrPut(key) { ArrayList<Int>() }\n list.add(element)\n }\n return destination}\n\n/**\n *
Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each
element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n *
@return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*/\npublic inline fun <K, M : MutableMap<in K, MutableList<Long>>> LongArray.groupByTo(destination: M,\n
keySelector: (Long) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list =\n
destination.getOrPut(key) { ArrayList<Long>() }\n list.add(element)\n }\n return destination}\n\n/**\n *
Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each
element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n *
@return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*/\npublic inline fun <K, M : MutableMap<in K, MutableList<Float>>> FloatArray.groupByTo(destination: M,\n
keySelector: (Float) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list =\n
destination.getOrPut(key) { ArrayList<Float>() }\n list.add(element)\n }\n return destination}\n\n/**\n *
Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each
element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n *
@return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*/\npublic inline fun <K, M : MutableMap<in K, MutableList<Double>>> DoubleArray.groupByTo(destination: M,\n
keySelector: (Double) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list =\n
destination.getOrPut(key) { ArrayList<Double>() }\n list.add(element)\n }\n return destination}\n\n/**\n *
Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each
element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n *
@return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*/\npublic inline fun <K, M : MutableMap<in K, MutableList<Boolean>>> BooleanArray.groupByTo(destination:\n
M, keySelector: (Boolean) -> K): M {\n for (element in this) {\n val key = keySelector(element)\n val list\n
= destination.getOrPut(key) { ArrayList<Boolean>() }\n list.add(element)\n }\n return\n
destination}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector]\n
function\n * applied to each element and puts to the [destination] map each group key associated with a list of\n
corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample\n
samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <K, M : MutableMap<in K,\n
MutableList<Char>>> CharArray.groupByTo(destination: M, keySelector: (Char) -> K): M {\n for (element in\n
this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<Char>() }\n\n
list.add(element)\n }\n return destination}\n\n/**\n * Groups values returned by the [valueTransform] function\n
applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to\n
the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n
*\n * \n * @return The [destination] map.\n * \n * @sample\n
samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <T, K, V, M :  

MutableMap<in K, MutableList<V>>> Array<out T>.groupByTo(destination: M, keySelector: (T) -> K,\n
valueTransform: (T) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list =\n
destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return\n
destination}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the\n
original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the\n
[destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]\n
map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic  

inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ByteArray.groupByTo(destination: M, keySelector:\n
(Byte) -> K, valueTransform: (Byte) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n

```

```

    val list = destination.getOrPut(key) { ArrayList<V>() } list.add(valueTransform(element)) } return
destination\n\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the
original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the
[destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]
map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ShortArray.groupByTo(destination: M, keySelector:
(Short) -> K, valueTransform: (Short) -> V): M {\n    for (element in this) {\n        val key = keySelector(element)\n
        val list = destination.getOrPut(key) { ArrayList<V>() } list.add(valueTransform(element))\n    } return
destination\n\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the
original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the
[destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]
map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> IntArray.groupByTo(destination: M, keySelector:
(Int) -> K, valueTransform: (Int) -> V): M {\n    for (element in this) {\n        val key = keySelector(element)\n
        val list = destination.getOrPut(key) { ArrayList<V>() } list.add(valueTransform(element))\n    } return
destination\n\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the
original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the
[destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]
map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> LongArray.groupByTo(destination: M, keySelector:
(Long) -> K, valueTransform: (Long) -> V): M {\n    for (element in this) {\n        val key = keySelector(element)\n
        val list = destination.getOrPut(key) { ArrayList<V>() } list.add(valueTransform(element))\n    } return
destination\n\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the
original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the
[destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]
map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> FloatArray.groupByTo(destination: M, keySelector:
(Float) -> K, valueTransform: (Float) -> V): M {\n    for (element in this) {\n        val key = keySelector(element)\n
        val list = destination.getOrPut(key) { ArrayList<V>() } list.add(valueTransform(element))\n    } return
destination\n\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the
original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the
[destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]
map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> DoubleArray.groupByTo(destination: M, keySelector:
(Double) -> K, valueTransform: (Double) -> V): M {\n    for (element in this) {\n        val key =
keySelector(element)\n        val list = destination.getOrPut(key) { ArrayList<V>() } list.add(valueTransform(element))\n    } return
destination\n\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given
[keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a
list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic inline fun <K, V, M :
MutableMap<in K, MutableList<V>>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K,
valueTransform: (Boolean) -> V): M {\n    for (element in this) {\n        val key = keySelector(element)\n        val list
= destination.getOrPut(key) { ArrayList<V>() } list.add(valueTransform(element))\n    } return
destination\n\n\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the
original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the
[destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]
map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic

```

```

inline fun <K, V, M : MutableMap<in K, MutableList<V>>> CharArray.groupByTo(destination: M, keySelector:
(Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
operations
 * using the specified [keySelector] function to extract a key from each element.
 * @sample
samples.collections.Grouping.groupingByEachCount
 * @since Kotlin("1.1")
public inline fun <T, K>
Array<out T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T,
K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun
keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a list containing the results of applying
the given [transform] function
 * to each element in the original array.
 * @sample
samples.collections.Collections.Transformations.map
 * @public inline fun <T, R> Array<out T>.map(transform:
(T) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the
results of applying the given [transform] function
 * to each element in the original array.
 * @sample
samples.collections.Collections.Transformations.map
 * @public inline fun <R> ByteArray.map(transform: (Byte)
-> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the
results of applying the given [transform] function
 * to each element in the original array.
 * @sample
samples.collections.Collections.Transformations.map
 * @public inline fun <R> ShortArray.map(transform:
(Short) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing
the results of applying the given [transform] function
 * to each element in the original array.
 * @sample
samples.collections.Collections.Transformations.map
 * @public inline fun <R> IntArray.map(transform: (Int) ->
R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results
of applying the given [transform] function
 * to each element in the original array.
 * @sample
samples.collections.Collections.Transformations.map
 * @public inline fun <R> LongArray.map(transform:
(Long) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing
the results of applying the given [transform] function
 * to each element in the original array.
 * @sample
samples.collections.Collections.Transformations.map
 * @public inline fun <R> FloatArray.map(transform: (Float)
-> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the
results of applying the given [transform] function
 * to each element in the original array.
 * @sample
samples.collections.Collections.Transformations.map
 * @public inline fun <R> DoubleArray.map(transform:
(Double) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing
the results of applying the given [transform] function
 * to each element in the original array.
 * @sample
samples.collections.Collections.Transformations.map
 * @public inline fun <R> BooleanArray.map(transform:
(Boolean) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list
containing the results of applying the given [transform] function
 * to each element in the original array.
 * @sample
samples.collections.Collections.Transformations.map
 * @public inline fun <R>
CharArray.map(transform: (Char) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in
the original array.
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 * @public inline fun <T, R> Array<out
T>.mapIndexed(transform: (index: Int, T) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size),
transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each
element and its index in the original array.
 * @param [transform] function that takes the index of an element and
the element itself
 * and returns the result of the transform applied to the element.
 * @public inline fun <R>
ByteArray.mapIndexed(transform: (index: Int, Byte) -> R): List<R> {
    return
mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the
given [transform] function
 * to each element and its index in the original array.
 * @param [transform] function
that takes the index of an element and the element itself
 * and returns the result of the transform applied to the
element.
 * @public inline fun <R> ShortArray.mapIndexed(transform: (index: Int, Short) -> R): List<R> {
}

```

```

return mapIndexedTo(ArrayList<R>(size), transform)\n\n/**\n * Returns a list containing the results of applying
the given [transform] function\n * to each element and its index in the original array.\n * @param [transform]
function that takes the index of an element and the element itself\n * and returns the result of the transform applied
to the element.\n */\npublic inline fun <R> IntArray.mapIndexed(transform: (index: Int, Int) -> R): List<R> {\n
return mapIndexedTo(ArrayList<R>(size), transform)\n\n/**\n * Returns a list containing the results of applying
the given [transform] function\n * to each element and its index in the original array.\n * @param [transform]
function that takes the index of an element and the element itself\n * and returns the result of the transform applied
to the element.\n */\npublic inline fun <R> LongArray.mapIndexed(transform: (index: Int, Long) -> R): List<R> {\n
return mapIndexedTo(ArrayList<R>(size), transform)\n\n/**\n * Returns a list containing the results of applying
the given [transform] function\n * to each element and its index in the original array.\n * @param [transform]
function that takes the index of an element and the element itself\n * and returns the result of the
transform applied to the element.\n */\npublic inline fun <R> FloatArray.mapIndexed(transform: (index: Int, Float) -
> R): List<R> {\n return mapIndexedTo(ArrayList<R>(size), transform)\n\n/**\n * Returns a list containing
the results of applying the given [transform] function\n * to each element and its index in the original array.\n *
@param [transform] function that takes the index of an element and the element itself\n * and returns the result of
the transform applied to the element.\n */\npublic inline fun <R> DoubleArray.mapIndexed(transform: (index: Int,
Double) -> R): List<R> {\n return mapIndexedTo(ArrayList<R>(size), transform)\n\n/**\n * Returns a list
containing the results of applying the given [transform] function\n * to each element and its index in the original
array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the
result of the transform applied to the element.\n */\npublic inline fun <R> BooleanArray.mapIndexed(transform:
(index: Int, Boolean) -> R): List<R> {\n return mapIndexedTo(ArrayList<R>(size), transform)\n\n/**\n *
Returns a list containing the results of applying the given [transform] function\n * to each element and its index in
the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and
returns the result of the transform applied to the element.\n */\npublic inline fun <R>
CharArray.mapIndexed(transform: (index: Int, Char) -> R): List<R> {\n return
mapIndexedTo(ArrayList<R>(size), transform)\n\n/**\n * Returns a list containing only the non-null results of
applying the given [transform] function\n * to each element and its index in the original array.\n * @param
[transform] function that takes the index of an element and the element itself\n * and returns the result of the
transform applied to the element.\n */\npublic inline fun <T, R : Any> Array<out
T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): List<R> {\n return
mapIndexedNotNullTo(ArrayList<R>(), transform)\n\n/**\n * Applies the given [transform] function to each
element and its index in the original array\n * and appends only the non-null results to the given [destination].\n *
@param [transform] function that takes the index of an element and the element itself\n * and returns the result of
the transform applied to the element.\n */\npublic inline fun <T, R : Any, C : MutableCollection<in R>> Array<out
T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {\n forEachIndexed { index,
element -> transform(index, element)?.let { destination.add(it) } }\n return destination\n\n/**\n * Applies the
given [transform] function to each element and its index in the original array\n * and appends the results to the given
[destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and
returns the result of the transform applied to the element.\n */\npublic inline fun <T, R, C : MutableCollection<in
R>> Array<out T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {\n var index = 0\n for
(item in this)\n destination.add(transform(index++, item))\n return destination\n\n/**\n * Applies the given
[transform] function to each element and its index in the original array\n * and appends the results to the given
[destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and
returns the result of the transform applied to the element.\n */\npublic inline fun <R, C : MutableCollection<in R>>
ByteArray.mapIndexedTo(destination: C, transform: (index: Int, Byte) -> R): C {\n var index = 0\n for (item in
this)\n destination.add(transform(index++, item))\n return destination\n\n/**\n * Applies the given
[transform] function to each element and its index in the original array\n * and appends the results to the given
[destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and

```


returns the result of the transform applied to the element.

```

public inline fun <R, C : MutableCollection<in R>>
ShortArray.mapIndexedTo(destination: C, transform: (index: Int, Short) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

```

* Applies the given [transform] function to each element and its index in the original array and appends the results to the given [destination].

@param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

```

public inline fun <R, C : MutableCollection<in R>>
IntArray.mapIndexedTo(destination: C, transform: (index: Int, Int) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

```

* Applies the given [transform] function to each element and its index in the original array and appends the results to the given [destination].

@param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

```

public inline fun <R, C : MutableCollection<in R>>
LongArray.mapIndexedTo(destination: C, transform: (index: Int, Long) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

```

* Applies the given [transform] function to each element and its index in the original array and appends the results to the given [destination].

@param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

```

public inline fun <R, C : MutableCollection<in R>>
FloatArray.mapIndexedTo(destination: C, transform: (index: Int, Float) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

```

* Applies the given [transform] function to each element and its index in the original array and appends the results to the given [destination].

@param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

```

public inline fun <R, C : MutableCollection<in R>>
DoubleArray.mapIndexedTo(destination: C, transform: (index: Int, Double) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

```

* Applies the given [transform] function to each element and its index in the original array and appends the results to the given [destination].

@param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

```

public inline fun <R, C : MutableCollection<in R>>
BooleanArray.mapIndexedTo(destination: C, transform: (index: Int, Boolean) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

```

* Applies the given [transform] function to each element and its index in the original array and appends the results to the given [destination].

@param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

```

public inline fun <R, C : MutableCollection<in R>>
CharArray.mapIndexedTo(destination: C, transform: (index: Int, Char) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

```

* Returns a list containing only the non-null results of applying the given [transform] function to each element in the original array.

```

public inline fun <T, R : Any> Array<out T>.mapNotNull(transform: (T) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(), transform)
}

```

* Applies the given [transform] function to each element in the original array and appends only the non-null results to the given [destination].

```

public inline fun <T, R : Any, C : MutableCollection<in R>> Array<out T>.mapNotNullTo(destination: C, transform: (T) -> R?): C {
    forEach { element -> transform(element)?.let { destination.add(it) }
    return destination
}

```

* Applies the given [transform] function to each element of the original array and appends the results to the given [destination].

```

public inline fun <T, R, C : MutableCollection<in R>> Array<out T>.mapTo(destination: C, transform: (T) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

```

* Applies the given [transform] function to each element of the original array and appends the results to the given [destination].

```

public inline fun <R, C : MutableCollection<in R>>
ByteArray.mapTo(destination: C, transform: (Byte) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

```

* Applies the given [transform] function to each element of the original array and appends the results to the given [destination].

```

public inline fun <R, C : MutableCollection<in R>> ShortArray.mapTo(destination: C, transform: (Short) -> R): C {
    for (item in this)

```

`destination.add(transform(item))` return destination
 Applies the given [transform] function to each element of the original array and appends the results to the given [destination].
`<R, C : MutableCollection<in R>> IntArray.mapTo(destination: C, transform: (Int) -> R): C` for (item in this) destination.add(transform(item)) return destination
 Applies the given [transform] function to each element of the original array and appends the results to the given [destination].
`<R, C : MutableCollection<in R>> LongArray.mapTo(destination: C, transform: (Long) -> R): C` for (item in this) destination.add(transform(item)) return destination
 Applies the given [transform] function to each element of the original array and appends the results to the given [destination].
`<R, C : MutableCollection<in R>> FloatArray.mapTo(destination: C, transform: (Float) -> R): C` for (item in this) destination.add(transform(item)) return destination
 Applies the given [transform] function to each element of the original array and appends the results to the given [destination].
`<R, C : MutableCollection<in R>> DoubleArray.mapTo(destination: C, transform: (Double) -> R): C` for (item in this) destination.add(transform(item)) return destination
 Applies the given [transform] function to each element of the original array and appends the results to the given [destination].
`<R, C : MutableCollection<in R>> BooleanArray.mapTo(destination: C, transform: (Boolean) -> R): C` for (item in this) destination.add(transform(item)) return destination
 Applies the given [transform] function to each element of the original array and appends the results to the given [destination].
`<R, C : MutableCollection<in R>> CharArray.mapTo(destination: C, transform: (Char) -> R): C` for (item in this) destination.add(transform(item)) return destination
 Returns a lazy [Iterable] that wraps each element of the original array into an [IndexedValue] containing the index of that element and the element itself.
`<T> Array<out T>.withIndex(): Iterable<IndexedValue<T>>` return IndexingIterable { iterator() }
 Returns a lazy [Iterable] that wraps each element of the original array into an [IndexedValue] containing the index of that element and the element itself.
`ByteArray.withIndex(): Iterable<IndexedValue<Byte>>` return IndexingIterable { iterator() }
 Returns a lazy [Iterable] that wraps each element of the original array into an [IndexedValue] containing the index of that element and the element itself.
`ShortArray.withIndex(): Iterable<IndexedValue<Short>>` return IndexingIterable { iterator() }
 Returns a lazy [Iterable] that wraps each element of the original array into an [IndexedValue] containing the index of that element and the element itself.
`IntArray.withIndex(): Iterable<IndexedValue<Int>>` return IndexingIterable { iterator() }
 Returns a lazy [Iterable] that wraps each element of the original array into an [IndexedValue] containing the index of that element and the element itself.
`LongArray.withIndex(): Iterable<IndexedValue<Long>>` return IndexingIterable { iterator() }
 Returns a lazy [Iterable] that wraps each element of the original array into an [IndexedValue] containing the index of that element and the element itself.
`FloatArray.withIndex(): Iterable<IndexedValue<Float>>` return IndexingIterable { iterator() }
 Returns a lazy [Iterable] that wraps each element of the original array into an [IndexedValue] containing the index of that element and the element itself.
`DoubleArray.withIndex(): Iterable<IndexedValue<Double>>` return IndexingIterable { iterator() }
 Returns a lazy [Iterable] that wraps each element of the original array into an [IndexedValue] containing the index of that element and the element itself.
`BooleanArray.withIndex(): Iterable<IndexedValue<Boolean>>` return IndexingIterable { iterator() }
 Returns a lazy [Iterable] that wraps each element of the original array into an [IndexedValue] containing the index of that element and the element itself.
`CharArray.withIndex(): Iterable<IndexedValue<Char>>` return IndexingIterable { iterator() }
 Returns a list containing only distinct elements from the given array. The elements in the resulting list are in the same order as they were in the source array.
`<T> Array<out T>.distinct(): List<T>` return this.toMutableSet().toList()
 Returns a list containing only distinct elements from the given array. The elements in the resulting list are in the same order as they were in the source array.
`ByteArray.distinct(): List<Byte>` return this.toMutableSet().toList()

only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic fun ShortArray.distinct(): List<Short> {\n return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic fun IntArray.distinct(): List<Int> {\n return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic fun LongArray.distinct(): List<Long> {\n return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic fun FloatArray.distinct(): List<Float> {\n return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic fun DoubleArray.distinct(): List<Double> {\n return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic fun BooleanArray.distinct(): List<Boolean> {\n return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only distinct elements from the given array.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic fun CharArray.distinct(): List<Char> {\n return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic inline fun <T, K> Array<out T>.distinctBy(selector: (T) -> K): List<T> {\n val set = HashSet<K>()\n val list = ArrayList<T>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic inline fun <K> ByteArray.distinctBy(selector: (Byte) -> K): List<Byte> {\n val set = HashSet<K>()\n val list = ArrayList<Byte>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic inline fun <K> ShortArray.distinctBy(selector: (Short) -> K): List<Short> {\n val set = HashSet<K>()\n val list = ArrayList<Short>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic inline fun <K> IntArray.distinctBy(selector: (Int) -> K): List<Int> {\n val set = HashSet<K>()\n val list = ArrayList<Int>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic inline fun <K> LongArray.distinctBy(selector: (Long) -> K): List<Long> {\n val set = HashSet<K>()\n val list = ArrayList<Long>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic inline fun <K> FloatArray.distinctBy(selector: (Float) -> K): List<Float> {\n val set = HashSet<K>()\n val list = ArrayList<Float>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic inline fun <K> DoubleArray.distinctBy(selector: (Double) -> K): List<Double> {\n val set = HashSet<K>()\n val list = ArrayList<Double>()\n for (e in this) {\n val key = selector(e)\n if (set.add(key))\n list.add(e)\n }\n return list\n}\n}

```

}\n  return list}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic inline fun <K> BooleanArray.distinctBy(selector: (Boolean) -> K): List<Boolean> {\n  val set = HashSet<K>()\n  val list = ArrayList<Boolean>()\n  for (e in this) {\n    val key = selector(e)\n    if (set.add(key))\n      list.add(e)\n  }\n  return list}\n\n/**\n * Returns a list containing only elements from the given array\n * having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the same order as they were in the source array.\n */\npublic inline fun <K> CharArray.distinctBy(selector: (Char) -> K): List<Char> {\n  val set = HashSet<K>()\n  val list = ArrayList<Char>()\n  for (e in this) {\n    val key = selector(e)\n    if (set.add(key))\n      list.add(e)\n  }\n  return list}\n\n/**\n * Returns a set containing all elements that are contained by both this set and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun <T> Array<out T>.intersect(other: Iterable<T>): Set<T> {\n  val set = this.toMutableSet()\n  set.retainAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by both this set and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun ByteArray.intersect(other: Iterable<Byte>): Set<Byte> {\n  val set = this.toMutableSet()\n  set.retainAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by both this set and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun ShortArray.intersect(other: Iterable<Short>): Set<Short> {\n  val set = this.toMutableSet()\n  set.retainAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by both this set and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun IntArray.intersect(other: Iterable<Int>): Set<Int> {\n  val set = this.toMutableSet()\n  set.retainAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by both this set and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun LongArray.intersect(other: Iterable<Long>): Set<Long> {\n  val set = this.toMutableSet()\n  set.retainAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by both this set and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun FloatArray.intersect(other: Iterable<Float>): Set<Float> {\n  val set = this.toMutableSet()\n  set.retainAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by both this set and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun DoubleArray.intersect(other: Iterable<Double>): Set<Double> {\n  val set = this.toMutableSet()\n  set.retainAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by both this set and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun BooleanArray.intersect(other: Iterable<Boolean>): Set<Boolean> {\n  val set = this.toMutableSet()\n  set.retainAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by both this set and the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun CharArray.intersect(other: Iterable<Char>): Set<Char> {\n  val set = this.toMutableSet()\n  set.retainAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun <T> Array<out T>.subtract(other: Iterable<T>): Set<T> {\n  val set = this.toMutableSet()\n  set.removeAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> {\n  val set = this.toMutableSet()\n  set.removeAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original array.\n */\npublic infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> {\n  val set = this.toMutableSet()\n  set.removeAll(other)\n  return set}\n\n/**\n * Returns a set containing all elements that are contained by this array and not contained by the specified collection.\n * \n * The returned set preserves the

```

element iteration order of the original array.

```

public infix fun IntArray.subtract(other: Iterable<Int>): Set<Int>
{
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

```

* Returns a set containing all elements that are contained by this array and not contained by the specified collection.

* The returned set preserves the element iteration order of the original array.

```

public infix fun LongArray.subtract(other:
Iterable<Long>): Set<Long>
{
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

```

* Returns a set containing all elements that are contained by this array and not contained by the specified collection.

* The returned set preserves the element iteration order of the original array.

```

public infix fun FloatArray.subtract(other: Iterable<Float>): Set<Float>
{
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

```

* Returns a set containing all elements that are contained by this array and not contained by the specified collection.

* The returned set preserves the element iteration order of the original array.

```

public infix fun DoubleArray.subtract(other: Iterable<Double>): Set<Double>
{
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

```

* Returns a set containing all elements that are contained by this array and not contained by the specified collection.

* The returned set preserves the element iteration order of the original array.

```

public infix fun BooleanArray.subtract(other: Iterable<Boolean>):
Set<Boolean>
{
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

```

* Returns a set containing all elements that are contained by this array and not contained by the specified collection.

* The returned set preserves the element iteration order of the original array.

```

public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char>
{
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

```

* Returns a mutable set containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

```

public fun <T> Array<out T>.toMutableSet(): MutableSet<T>
{
    val set = LinkedHashSet<T>(mapCapacity(size))
    for (item in this) set.add(item)
    return set
}

```

* Returns a mutable set containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

```

public fun ByteArray.toMutableSet(): MutableSet<Byte>
{
    val set = LinkedHashSet<Byte>(mapCapacity(size))
    for (item in this) set.add(item)
    return set
}

```

* Returns a mutable set containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

```

public fun ShortArray.toMutableSet(): MutableSet<Short>
{
    val set = LinkedHashSet<Short>(mapCapacity(size))
    for (item in this) set.add(item)
    return set
}

```

* Returns a mutable set containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

```

public fun IntArray.toMutableSet(): MutableSet<Int>
{
    val set = LinkedHashSet<Int>(mapCapacity(size))
    for (item in this) set.add(item)
    return set
}

```

* Returns a mutable set containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

```

public fun LongArray.toMutableSet(): MutableSet<Long>
{
    val set = LinkedHashSet<Long>(mapCapacity(size))
    for (item in this) set.add(item)
    return set
}

```

* Returns a mutable set containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

```

public fun FloatArray.toMutableSet(): MutableSet<Float>
{
    val set = LinkedHashSet<Float>(mapCapacity(size))
    for (item in this) set.add(item)
    return set
}

```

* Returns a mutable set containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

```

public fun DoubleArray.toMutableSet(): MutableSet<Double>
{
    val set = LinkedHashSet<Double>(mapCapacity(size))
    for (item in this) set.add(item)
    return set
}

```

* Returns a mutable set containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

```

public fun BooleanArray.toMutableSet(): MutableSet<Boolean>
{
    val set = LinkedHashSet<Boolean>(mapCapacity(size))
    for (item in this) set.add(item)
    return set
}

```

* Returns a mutable set containing all distinct elements from the given array.

* The returned set preserves the element iteration order of the original array.

```

public fun CharArray.toMutableSet(): MutableSet<Char>
{
    val set = LinkedHashSet<Char>(mapCapacity(size))
    for (item in this) set.add(item)
    return set
}

```

* Returns a set containing all distinct elements from both collections.

* The returned set preserves the element iteration order of the original array.

* Those elements of the [other] collection that are unique are iterated in the

```

end\n * in the order of the [other] collection.\n *\npublic infix fun <T> Array<out T>.union(other: Iterable<T>):
Set<T> {\n  val set = this.toMutableSet()\n  set.addAll(other)\n  return set\n}\n\n/**\n * Returns a set containing
all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the
original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of
the [other] collection.\n *\npublic infix fun ByteArray.union(other: Iterable<Byte>): Set<Byte> {\n  val set =
this.toMutableSet()\n  set.addAll(other)\n  return set\n}\n\n/**\n * Returns a set containing all distinct elements
from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those
elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n
*\npublic infix fun ShortArray.union(other: Iterable<Short>): Set<Short> {\n  val set = this.toMutableSet()\n
set.addAll(other)\n  return set\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n
*\n * The returned set preserves the element iteration order of the original array.\n * Those elements of the [other]
collection that are unique are iterated in the end\n * in the order of the [other] collection.\n *\npublic infix fun
IntArray.union(other: Iterable<Int>): Set<Int> {\n  val set = this.toMutableSet()\n  set.addAll(other)\n  return
set\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set
preserves the element iteration order of the original array.\n * Those elements of the [other] collection that are
unique are iterated in the end\n * in the order of the [other] collection.\n *\npublic infix fun LongArray.union(other:
Iterable<Long>): Set<Long> {\n  val set = this.toMutableSet()\n  set.addAll(other)\n  return set\n}\n\n/**\n *
Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element
iteration order of the original array.\n * Those elements of the [other] collection that are unique are iterated in the
end\n * in the order of the [other] collection.\n *\npublic infix fun FloatArray.union(other: Iterable<Float>):
Set<Float> {\n  val set = this.toMutableSet()\n  set.addAll(other)\n  return set\n}\n\n/**\n * Returns a set
containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order
of the original array.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the
order of the [other] collection.\n *\npublic infix fun DoubleArray.union(other: Iterable<Double>): Set<Double> {\n
  val set = this.toMutableSet()\n  set.addAll(other)\n  return set\n}\n\n/**\n * Returns a set containing all distinct
elements from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n
*\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other]
collection.\n *\npublic infix fun BooleanArray.union(other: Iterable<Boolean>): Set<Boolean> {\n  val set =
this.toMutableSet()\n  set.addAll(other)\n  return set\n}\n\n/**\n * Returns a set containing all distinct elements
from both collections.\n * \n * The returned set preserves the element iteration order of the original array.\n * Those
elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n
*\npublic infix fun CharArray.union(other: Iterable<Char>): Set<Char> {\n  val set = this.toMutableSet()\n
set.addAll(other)\n  return set\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n *
@sample samples.collections.Collections.Aggregates.all\n *\npublic inline fun <T> Array<out T>.all(predicate: (T)
-> Boolean): Boolean {\n  for (element in this) if (!predicate(element)) return false\n  return true\n}\n\n/**\n *
Returns `true` if all elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.all\n *\npublic inline fun ByteArray.all(predicate: (Byte) -> Boolean):
Boolean {\n  for (element in this) if (!predicate(element)) return false\n  return true\n}\n\n/**\n * Returns `true`
if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n
*\npublic inline fun ShortArray.all(predicate: (Short) -> Boolean): Boolean {\n  for (element in this) if
(!predicate(element)) return false\n  return true\n}\n\n/**\n * Returns `true` if all elements match the given
[predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n *\npublic inline fun
IntArray.all(predicate: (Int) -> Boolean): Boolean {\n  for (element in this) if (!predicate(element)) return false\n
return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.all\n *\npublic inline fun LongArray.all(predicate: (Long) -> Boolean):
Boolean {\n  for (element in this) if (!predicate(element)) return false\n  return true\n}\n\n/**\n * Returns `true`
if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n
*\npublic inline fun FloatArray.all(predicate: (Float) -> Boolean): Boolean {\n  for (element in this) if

```

```

(!predicate(element)) return false\n    return true\n}\n\n/**\n * Returns `true` if all elements match the given
[predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun
DoubleArray.all(predicate: (Double) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return
false\n    return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.all\n */\npublic inline fun BooleanArray.all(predicate: (Boolean) ->
Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}\n\n/**\n *
Returns `true` if all elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.all\n */\npublic inline fun CharArray.all(predicate: (Char) -> Boolean):
Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}\n\n/**\n * Returns `true` if
array has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun
<T> Array<out T>.any(): Boolean {\n    return !isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one
element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun ByteArray.any():
Boolean {\n    return !isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic fun ShortArray.any(): Boolean {\n    return
!isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic fun IntArray.any(): Boolean {\n    return
!isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic fun LongArray.any(): Boolean {\n    return
!isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic fun FloatArray.any(): Boolean {\n    return
!isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic fun DoubleArray.any(): Boolean {\n    return
!isEmpty()\n}\n\n/**\n * Returns `true` if array has at least one element.\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic fun BooleanArray.any(): Boolean {\n    return
!isEmpty()\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic inline fun <T> Array<out
T>.any(predicate: (T) -> Boolean): Boolean {\n    for (element in this) if (predicate(element)) return true\n    return
false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic inline fun ByteArray.any(predicate:
(Byte) -> Boolean): Boolean {\n    for (element in this) if (predicate(element)) return true\n    return
false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic inline fun ShortArray.any(predicate:
(Short) -> Boolean): Boolean {\n    for (element in this) if (predicate(element)) return true\n    return
false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic inline fun IntArray.any(predicate: (Int) -
> Boolean): Boolean {\n    for (element in this) if (predicate(element)) return true\n    return false\n}\n\n/**\n *
Returns `true` if at least one element matches the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic inline fun LongArray.any(predicate:
(Long) -> Boolean): Boolean {\n    for (element in this) if (predicate(element)) return true\n    return
false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic inline fun FloatArray.any(predicate:
(Float) -> Boolean): Boolean {\n    for (element in this) if (predicate(element)) return true\n    return
false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.any\n */\npublic inline fun DoubleArray.any(predicate:
(Double) -> Boolean): Boolean {\n    for (element in this) if (predicate(element)) return true\n    return
false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample

```

```

samples.collections.Collections.Aggregates.anyWithPredicate\n *^\npublic inline fun BooleanArray.any(predicate:
(Boolean) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return true\n  return
false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.anyWithPredicate\n *^\npublic inline fun CharArray.any(predicate:
(Char) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return true\n  return
false\n}\n\n/**\n * Returns the number of elements in this array.\n *^\n@kotlin.internal.InlineOnly\npublic inline
fun <T> Array<out T>.count(): Int {\n  return size\n}\n\n/**\n * Returns the number of elements in this array.\n
*^\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.count(): Int {\n  return size\n}\n\n/**\n * Returns the
number of elements in this array.\n *^\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.count(): Int {\n
return size\n}\n\n/**\n * Returns the number of elements in this array.\n *^\n@kotlin.internal.InlineOnly\npublic
inline fun IntArray.count(): Int {\n  return size\n}\n\n/**\n * Returns the number of elements in this array.\n
*^\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.count(): Int {\n  return size\n}\n\n/**\n * Returns the
number of elements in this array.\n *^\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.count(): Int {\n
return size\n}\n\n/**\n * Returns the number of elements in this array.\n *^\n@kotlin.internal.InlineOnly\npublic
inline fun DoubleArray.count(): Int {\n  return size\n}\n\n/**\n * Returns the number of elements in this array.\n
*^\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.count(): Int {\n  return size\n}\n\n/**\n * Returns
the number of elements in this array.\n *^\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.count(): Int {\n
return size\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n *^\npublic inline fun
<T> Array<out T>.count(predicate: (T) -> Boolean): Int {\n  var count = 0\n  for (element in this) if
(predicate(element)) ++count\n  return count\n}\n\n/**\n * Returns the number of elements matching the given
[predicate].\n *^\npublic inline fun ByteArray.count(predicate: (Byte) -> Boolean): Int {\n  var count = 0\n  for
(element in this) if (predicate(element)) ++count\n  return count\n}\n\n/**\n * Returns the number of elements
matching the given [predicate].\n *^\npublic inline fun ShortArray.count(predicate: (Short) -> Boolean): Int {\n  var
count = 0\n  for (element in this) if (predicate(element)) ++count\n  return count\n}\n\n/**\n * Returns the
number of elements matching the given [predicate].\n *^\npublic inline fun IntArray.count(predicate: (Int) ->
Boolean): Int {\n  var count = 0\n  for (element in this) if (predicate(element)) ++count\n  return
count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n *^\npublic inline fun
LongArray.count(predicate: (Long) -> Boolean): Int {\n  var count = 0\n  for (element in this) if
(predicate(element)) ++count\n  return count\n}\n\n/**\n * Returns the number of elements matching the given
[predicate].\n *^\npublic inline fun FloatArray.count(predicate: (Float) -> Boolean): Int {\n  var count = 0\n  for
(element in this) if (predicate(element)) ++count\n  return count\n}\n\n/**\n * Returns the number of elements
matching the given [predicate].\n *^\npublic inline fun DoubleArray.count(predicate: (Double) -> Boolean): Int {\n
var count = 0\n  for (element in this) if (predicate(element)) ++count\n  return count\n}\n\n/**\n * Returns the
number of elements matching the given [predicate].\n *^\npublic inline fun BooleanArray.count(predicate: (Boolean)
-> Boolean): Int {\n  var count = 0\n  for (element in this) if (predicate(element)) ++count\n  return
count\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n *^\npublic inline fun
CharArray.count(predicate: (Char) -> Boolean): Int {\n  var count = 0\n  for (element in this) if
(predicate(element)) ++count\n  return count\n}\n\n/**\n * Accumulates value starting with [initial] value and
applying [operation] from left to right to current accumulator value and each element.\n *^\npublic inline fun <T, R>
Array<out T>.fold(initial: R, operation: (acc: R, T) -> R): R {\n  var accumulator = initial\n  for (element in this)
accumulator = operation(accumulator, element)\n  return accumulator\n}\n\n/**\n * Accumulates value starting
with [initial] value and applying [operation] from left to right to current accumulator value and each element.\n
*^\npublic inline fun <R> ByteArray.fold(initial: R, operation: (acc: R, Byte) -> R): R {\n  var accumulator =
initial\n  for (element in this) accumulator = operation(accumulator, element)\n  return accumulator\n}\n\n/**\n
* Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator
value and each element.\n *^\npublic inline fun <R> ShortArray.fold(initial: R, operation: (acc: R, Short) -> R): R
{\n  var accumulator = initial\n  for (element in this) accumulator = operation(accumulator, element)\n  return
accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right

```



```

to current accumulator value and each element.\n */\npublic inline fun <R> IntArray.fold(initial: R, operation: (acc:
R, Int) -> R): R {\n    var accumulator = initial\n    for (element in this) accumulator = operation(accumulator,
element)\n    return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying
[operation] from left to right to current accumulator value and each element.\n */\npublic inline fun <R>
LongArray.fold(initial: R, operation: (acc: R, Long) -> R): R {\n    var accumulator = initial\n    for (element in this)
accumulator = operation(accumulator, element)\n    return accumulator\n}\n\n/**\n * Accumulates value starting
with [initial] value and applying [operation] from left to right to current accumulator value and each element.\n
*/\npublic inline fun <R> FloatArray.fold(initial: R, operation: (acc: R, Float) -> R): R {\n    var accumulator =
initial\n    for (element in this) accumulator = operation(accumulator, element)\n    return accumulator\n}\n\n/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator
value and each element.\n */\npublic inline fun <R> DoubleArray.fold(initial: R, operation: (acc: R, Double) -> R):
R {\n    var accumulator = initial\n    for (element in this) accumulator = operation(accumulator, element)\n    return
accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right
to current accumulator value and each element.\n */\npublic inline fun <R> BooleanArray.fold(initial: R, operation:
(acc: R, Boolean) -> R): R {\n    var accumulator = initial\n    for (element in this) accumulator =
operation(accumulator, element)\n    return accumulator\n}\n\n/**\n * Accumulates value starting with [initial]
value and applying [operation] from left to right to current accumulator value and each element.\n */\npublic inline
fun <R> CharArray.fold(initial: R, operation: (acc: R, Char) -> R): R {\n    var accumulator = initial\n    for (element
in this) accumulator = operation(accumulator, element)\n    return accumulator\n}\n\n/**\n * Accumulates value
starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each
element with its index in the original array.\n * @param [operation] function that takes the index of an element,
current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline
fun <T, R> Array<out T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {\n    var index = 0\n    var
accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return
accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left
to right\n * to current accumulator value and each element with its index in the original array.\n * @param
[operation] function that takes the index of an element, current accumulator value\n * and the element itself, and
calculates the next accumulator value.\n */\npublic inline fun <R> ByteArray.foldIndexed(initial: R, operation:
(index: Int, acc: R, Byte) -> R): R {\n    var index = 0\n    var accumulator = initial\n    for (element in this)
accumulator = operation(index++, accumulator, element)\n    return accumulator\n}\n\n/**\n * Accumulates value
starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each
element with its index in the original array.\n * @param [operation] function that takes the index of an element,
current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline
fun <R> ShortArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): R {\n    var index = 0\n    var
accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return
accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation]
function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the
next accumulator value.\n */\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R,
Int) -> R): R {\n    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator =
operation(index++, accumulator, element)\n    return accumulator\n}\n\n/**\n * Accumulates value starting with
[initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its
index in the original array.\n * @param [operation] function that takes the index of an element, current accumulator
value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <R>
LongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Long) -> R): R {\n    var index = 0\n    var
accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return
accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation]

```

function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

```

public inline fun <R> FloatArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

@param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

```

public inline fun <R> DoubleArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

@param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

```

public inline fun <R> BooleanArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

@param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

```

public inline fun <R> CharArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.

```

public inline fun <T, R> Array<out T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.

```

public inline fun <R> ByteArray.foldRight(initial: R, operation: (Byte, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.

```

public inline fun <R> ShortArray.foldRight(initial: R, operation: (Short, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.

```

public inline fun <R> IntArray.foldRight(initial: R, operation: (Int, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.

```

public inline fun <R> LongArray.foldRight(initial: R, operation: (Long, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.

```

public inline fun <R> FloatArray.foldRight(initial: R, operation: (Float, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.

```

public inline fun <R> DoubleArray.foldRight(initial: R, operation: (Double, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.

```

public inline fun <R> BooleanArray.foldRight(initial: R, operation: (Boolean, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

```

```

index = lastIndex\n  var accumulator = initial\n  while (index >= 0) {\n    accumulator = operation(get(index--),
accumulator)\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying
[operation] from right to left to each element and current accumulator value.\n */\npublic inline fun <R>
CharArray.foldRight(initial: R, operation: (Char, acc: R) -> R): R {\n  var index = lastIndex\n  var accumulator =
initial\n  while (index >= 0) {\n    accumulator = operation(get(index--), accumulator)\n  }\n  return
accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to
left\n * to each element with its index in the original array and current accumulator value.\n * @param [operation]
function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the
next accumulator value.\n */\npublic inline fun <T, R> Array<out T>.foldRightIndexed(initial: R, operation: (index:
Int, T, acc: R) -> R): R {\n  var index = lastIndex\n  var accumulator = initial\n  while (index >= 0) {\n
accumulator = operation(index, get(index), accumulator)\n    --index\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with
its index in the original array and current accumulator value.\n * @param [operation] function that takes the index of
an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> ByteArray.foldRightIndexed(initial: R, operation: (index: Int, Byte, acc: R) -> R): R {\n
var index = lastIndex\n  var accumulator = initial\n  while (index >= 0) {\n    accumulator = operation(index,
get(index), accumulator)\n    --index\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with
[initial] value and applying [operation] from right to left\n * to each element with its index in the original array and
current accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n *
and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R>
ShortArray.foldRightIndexed(initial: R, operation: (index: Int, Short, acc: R) -> R): R {\n  var index = lastIndex\n
var accumulator = initial\n  while (index >= 0) {\n    accumulator = operation(index, get(index), accumulator)\n
    --index\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying
[operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n
* @param [operation] function that takes the index of an element, the element itself\n * and current accumulator
value, and calculates the next accumulator value.\n */\npublic inline fun <R> IntArray.foldRightIndexed(initial: R,
operation: (index: Int, Int, acc: R) -> R): R {\n  var index = lastIndex\n  var accumulator = initial\n  while (index
>= 0) {\n    accumulator = operation(index, get(index), accumulator)\n    --index\n  }\n  return
accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to
left\n * to each element with its index in the original array and current accumulator value.\n * @param [operation]
function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the
next accumulator value.\n */\npublic inline fun <R> LongArray.foldRightIndexed(initial: R, operation: (index: Int,
Long, acc: R) -> R): R {\n  var index = lastIndex\n  var accumulator = initial\n  while (index >= 0) {\n
accumulator = operation(index, get(index), accumulator)\n    --index\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with
its index in the original array and current accumulator value.\n * @param [operation] function that takes the index of
an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R> FloatArray.foldRightIndexed(initial: R, operation: (index: Int, Float, acc: R) -> R): R {\n
var index = lastIndex\n  var accumulator = initial\n  while (index >= 0) {\n    accumulator = operation(index,
get(index), accumulator)\n    --index\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with
[initial] value and applying [operation] from right to left\n * to each element with its index in the original array and
current accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n *
and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R>
DoubleArray.foldRightIndexed(initial: R, operation: (index: Int, Double, acc: R) -> R): R {\n  var index =
lastIndex\n  var accumulator = initial\n  while (index >= 0) {\n    accumulator = operation(index, get(index),
accumulator)\n    --index\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with [initial]
value and applying [operation] from right to left\n * to each element with its index in the original array and current
accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n * and

```

```

current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R>
BooleanArray.foldRightIndexed(initial: R, operation: (index: Int, Boolean, acc: R) -> R): R {\n  var index =
lastIndex\n  var accumulator = initial\n  while (index >= 0) {\n    accumulator = operation(index, get(index),
accumulator)\n    --index\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with [initial]
value and applying [operation] from right to left\n * to each element with its index in the original array and current
accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n * and
current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <R>
CharArray.foldRightIndexed(initial: R, operation: (index: Int, Char, acc: R) -> R): R {\n  var index = lastIndex\n
var accumulator = initial\n  while (index >= 0) {\n    accumulator = operation(index, get(index), accumulator)\n
--index\n  }\n  return accumulator\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic
inline fun <T> Array<out T>.forEach(action: (T) -> Unit): Unit {\n  for (element in this)
action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun
ByteArray.forEach(action: (Byte) -> Unit): Unit {\n  for (element in this) action(element)\n}\n\n/**\n * Performs
the given [action] on each element.\n */\npublic inline fun ShortArray.forEach(action: (Short) -> Unit): Unit {\n
for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline
fun IntArray.forEach(action: (Int) -> Unit): Unit {\n  for (element in this) action(element)\n}\n\n/**\n * Performs
the given [action] on each element.\n */\npublic inline fun LongArray.forEach(action: (Long) -> Unit): Unit {\n
for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline
fun FloatArray.forEach(action: (Float) -> Unit): Unit {\n  for (element in this) action(element)\n}\n\n/**\n *
Performs the given [action] on each element.\n */\npublic inline fun DoubleArray.forEach(action: (Double) ->
Unit): Unit {\n  for (element in this) action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n
*/\npublic inline fun BooleanArray.forEach(action: (Boolean) -> Unit): Unit {\n  for (element in this)
action(element)\n}\n\n/**\n * Performs the given [action] on each element.\n */\npublic inline fun
CharArray.forEach(action: (Char) -> Unit): Unit {\n  for (element in this) action(element)\n}\n\n/**\n * Performs
the given [action] on each element, providing sequential index with the element.\n * @param [action] function that
takes the index of an element and the element itself\n * and performs the desired action on the element.\n */\npublic
inline fun <T> Array<out T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {\n  var index = 0\n  for (item
in this) action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index
with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and
performs the desired action on the element.\n */\npublic inline fun ByteArray.forEachIndexed(action: (index: Int,
Byte) -> Unit): Unit {\n  var index = 0\n  for (item in this) action(index++, item)\n}\n\n/**\n * Performs the
given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes
the index of an element and the element itself\n * and performs the desired action on the element.\n */\npublic inline
fun ShortArray.forEachIndexed(action: (index: Int, Short) -> Unit): Unit {\n  var index = 0\n  for (item in this)
action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with
the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs
the desired action on the element.\n */\npublic inline fun IntArray.forEachIndexed(action: (index: Int, Int) ->
Unit): Unit {\n  var index = 0\n  for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action]
on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an
element and the element itself\n * and performs the desired action on the element.\n */\npublic inline fun
LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {\n  var index = 0\n  for (item in this)
action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with
the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs
the desired action on the element.\n */\npublic inline fun FloatArray.forEachIndexed(action: (index: Int, Float) ->
Unit): Unit {\n  var index = 0\n  for (item in this) action(index++, item)\n}\n\n/**\n * Performs the given [action]
on each element, providing sequential index with the element.\n * @param [action] function that takes the index of
an element and the element itself\n * and performs the desired action on the element.\n */\npublic inline fun
DoubleArray.forEachIndexed(action: (index: Int, Double) -> Unit): Unit {\n  var index = 0\n  for (item in this)

```

```

action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with
the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs
the desired action on the element.\n */\npublic inline fun BooleanArray.forEachIndexed(action: (index: Int,
Boolean) -> Unit): Unit {\n    var index = 0\n    for (item in this) action(index++, item)\n}\n\n/**\n * Performs the
given [action] on each element, providing sequential index with the element.\n * @param [action] function that
takes the index of an element and the element itself\n * and performs the desired action on the element.\n */\npublic
inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {\n    var index = 0\n    for (item in
this) action(index++, item)\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n * \n * If
any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin("1.1")\npublic fun Array<out Double>.max():
Double? {\n    if (isEmpty()) return null\n    var max = this[0]\n    if (max.isNaN()) return max\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        if (e.isNaN()) return e\n        if (max < e) max = e\n    }\n    return
max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n * \n * If any of elements is `NaN`
returns `NaN`.\n */\n@SinceKotlin("1.1")\npublic fun Array<out Float>.max(): Float? {\n    if (isEmpty()) return
null\n    var max = this[0]\n    if (max.isNaN()) return max\n    for (i in 1..lastIndex) {\n        val e = this[i]\n
        if (e.isNaN()) return e\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the largest element or
`null` if there are no elements.\n */\npublic fun <T : Comparable<T>> Array<out T>.max(): T? {\n    if (isEmpty())
return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (max < e) max = e\n
    }\n    return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\npublic fun
ByteArray.max(): Byte? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e
= this[i]\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the largest element or `null` if there
are no elements.\n */\npublic fun ShortArray.max(): Short? {\n    if (isEmpty()) return null\n    var max = this[0]\n
for (i in 1..lastIndex) {\n        val e = this[i]\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns
the largest element or `null` if there are no elements.\n */\npublic fun IntArray.max(): Int? {\n    if (isEmpty()) return
null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (max < e) max = e\n
    }\n    return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\npublic fun
LongArray.max(): Long? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e
= this[i]\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n
* \n * If any of elements is `NaN` returns `NaN`.\n */\npublic fun FloatArray.max(): Float? {\n    if (isEmpty())
return null\n    var max = this[0]\n    if (max.isNaN()) return max\n    for (i in 1..lastIndex) {\n        val e = this[i]\n
        if (e.isNaN()) return e\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the largest element or
`null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\npublic fun
DoubleArray.max(): Double? {\n    if (isEmpty()) return null\n    var max = this[0]\n    if (max.isNaN()) return
max\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (e.isNaN()) return e\n        if (max < e) max = e\n
    }\n    return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n */\npublic fun
CharArray.max(): Char? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e
= this[i]\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element yielding the largest
value of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.maxBy\n */\npublic inline fun <T, R : Comparable<R>> Array<out
T>.maxBy(selector: (T) -> R): T? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex =
this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n
            maxValue = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value
of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.maxBy\n */\npublic inline fun <R : Comparable<R>>
ByteArray.maxBy(selector: (Byte) -> R): Byte? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i
in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n
            maxValue = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value

```

```

of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.maxBy\n */\npublic inline fun <R : Comparable<R>>
ShortArray.maxBy(selector: (Short) -> R): Short? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxV = selector(maxElem)\n    for (i
in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxV < v) {\n            maxElem = e\n
            maxV = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value
of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.maxBy\n */\npublic inline fun <R : Comparable<R>>
IntArray.maxBy(selector: (Int) -> R): Int? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxV = selector(maxElem)\n    for (i
in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxV < v) {\n            maxElem = e\n
            maxV = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value
of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.maxBy\n */\npublic inline fun <R : Comparable<R>>
LongArray.maxBy(selector: (Long) -> R): Long? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxV = selector(maxElem)\n    for (i
in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxV < v) {\n            maxElem = e\n
            maxV = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value
of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.maxBy\n */\npublic inline fun <R : Comparable<R>>
FloatArray.maxBy(selector: (Float) -> R): Float? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxV = selector(maxElem)\n    for (i
in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxV < v) {\n            maxElem = e\n
            maxV = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value
of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.maxBy\n */\npublic inline fun <R : Comparable<R>>
DoubleArray.maxBy(selector: (Double) -> R): Double? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n
val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxV = selector(maxElem)\n    for
(i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxV < v) {\n            maxElem = e\n
            maxV = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element yielding the largest
value of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.maxBy\n */\npublic inline fun <R : Comparable<R>>
BooleanArray.maxBy(selector: (Boolean) -> R): Boolean? {\n    if (isEmpty()) return null\n    var maxElem =
this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxV =
selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxV < v)
{\n            maxElem = e\n            maxV = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first
element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.maxBy\n */\npublic inline fun <R : Comparable<R>>
CharArray.maxBy(selector: (Char) -> R): Char? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxV = selector(maxElem)\n    for (i
in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxV < v) {\n            maxElem = e\n
            maxV = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element having the largest value
according to the provided [comparator] or `null` if there are no elements.\n */\n\npublic fun <T> Array<out
T>.maxWith(comparator: Comparator<in T>): T? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n    return
max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null`
if there are no elements.\n */\n\npublic fun ByteArray.maxWith(comparator: Comparator<in Byte>): Byte? {\n    if
(isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if

```

```

(comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element having the
largest value according to the provided [comparator] or `null` if there are no elements.\n */\npublic fun
ShortArray.maxWith(comparator: Comparator<in Short>): Short? {\n    if (isEmpty()) return null\n    var max =
this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator]
or `null` if there are no elements.\n */\npublic fun IntArray.maxWith(comparator: Comparator<in Int>): Int? {\n    if
(isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if
(comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\npublic fun
LongArray.maxWith(comparator: Comparator<in Long>): Long? {\n    if (isEmpty()) return null\n    var max =
this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator]
or `null` if there are no elements.\n */\npublic fun FloatArray.maxWith(comparator: Comparator<in Float>): Float?
{\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if
(comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\npublic fun
DoubleArray.maxWith(comparator: Comparator<in Double>): Double? {\n    if (isEmpty()) return null\n    var max
= this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator]
or `null` if there are no elements.\n */\npublic fun BooleanArray.maxWith(comparator: Comparator<in Boolean>):
Boolean? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\npublic fun
CharArray.maxWith(comparator: Comparator<in Char>): Char? {\n    if (isEmpty()) return null\n    var max =
this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n * If any of elements
is `NaN` returns `NaN`.\n */\n@SinceKotlin("1.1")\npublic fun Array<out Double>.min(): Double? {\n    if
(isEmpty()) return null\n    var min = this[0]\n    if (min.isNaN()) return min\n    for (i in 1..lastIndex) {\n        val e =
this[i]\n        if (e.isNaN()) return e\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n * If any of elements is `NaN` returns `NaN`.\n */\n@SinceKotlin("1.1")\npublic fun Array<out Float>.min(): Float? {\n    if (isEmpty()) return null\n    var min =
this[0]\n    if (min.isNaN()) return min\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (e.isNaN()) return e\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no
elements.\n */\npublic fun <T : Comparable<T>> Array<out T>.min(): T? {\n    if (isEmpty()) return null\n    var
min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (min > e) min = e\n    }\n    return
min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\npublic fun ByteArray.min():
Byte? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if
(min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no
elements.\n */\npublic fun ShortArray.min(): Short? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\npublic fun IntArray.min(): Int? {\n    if (isEmpty()) return null\n    var
min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (min > e) min = e\n    }\n    return
min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\npublic fun LongArray.min():
Long? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if
(min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no
elements.\n */\n * If any of elements is `NaN` returns `NaN`.\n */\npublic fun FloatArray.min(): Float? {\n    if
(isEmpty()) return null\n    var min = this[0]\n    if (min.isNaN()) return min\n    for (i in 1..lastIndex) {\n        val e =
this[i]\n        if (e.isNaN()) return e\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the

```

```

smallest element or `null` if there are no elements.\n * \n * If any of elements is `NaN` returns `NaN`.\n */\npublic
fun DoubleArray.min(): Double? {\n    if (isEmpty()) return null\n    var min = this[0]\n    if (min.isNaN()) return
min\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (e.isNaN()) return e\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\npublic fun
CharArray.min(): Char? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e
= this[i]\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the first element yielding the smallest
value of the given function or `null` if there are no elements.\n */\n * @sample
samples.collections.Collections.Aggregates.minBy\n */\npublic inline fun <T, R : Comparable<R>> Array<out
T>.minBy(selector: (T) -> R): T? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex =
this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n
minValue = v\n        }\n    }\n    return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value
of the given function or `null` if there are no elements.\n */\n * @sample
samples.collections.Collections.Aggregates.minBy\n */\npublic inline fun <R : Comparable<R>>
ByteArray.minBy(selector: (Byte) -> R): Byte? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n
minValue = v\n        }\n    }\n    return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value
of the given function or `null` if there are no elements.\n */\n * @sample
samples.collections.Collections.Aggregates.minBy\n */\npublic inline fun <R : Comparable<R>>
ShortArray.minBy(selector: (Short) -> R): Short? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n
minValue = v\n        }\n    }\n    return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value
of the given function or `null` if there are no elements.\n */\n * @sample
samples.collections.Collections.Aggregates.minBy\n */\npublic inline fun <R : Comparable<R>>
IntArray.minBy(selector: (Int) -> R): Int? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex
= this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n
minValue = v\n        }\n    }\n    return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value
of the given function or `null` if there are no elements.\n */\n * @sample
samples.collections.Collections.Aggregates.minBy\n */\npublic inline fun <R : Comparable<R>>
LongArray.minBy(selector: (Long) -> R): Long? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n
minValue = v\n        }\n    }\n    return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value
of the given function or `null` if there are no elements.\n */\n * @sample
samples.collections.Collections.Aggregates.minBy\n */\npublic inline fun <R : Comparable<R>>
FloatArray.minBy(selector: (Float) -> R): Float? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n
minValue = v\n        }\n    }\n    return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value
of the given function or `null` if there are no elements.\n */\n * @sample
samples.collections.Collections.Aggregates.minBy\n */\npublic inline fun <R : Comparable<R>>
DoubleArray.minBy(selector: (Double) -> R): Double? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n
val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i
in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n
minValue = v\n        }\n    }\n    return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value
of the given function or `null` if there are no elements.\n */\n * @sample
samples.collections.Collections.Aggregates.minBy\n */\npublic inline fun <R : Comparable<R>>

```


of the given function or `null` if there are no elements.

```

\n * \n * @sample
samples.collections.Collections.Aggregates.minBy\n * \n public inline fun <R : Comparable<R>>
BooleanArray.minBy(selector: (Boolean) -> R): Boolean? {\n if (isEmpty()) return null\n var minElem =
this[0]\n val lastIndex = this.lastIndex\n if (lastIndex == 0) return minElem\n var minValue =
selector(minElem)\n for (i in 1..lastIndex) {\n val e = this[i]\n val v = selector(e)\n if (minValue > v)
{\n minElem = e\n minValue = v\n }\n }\n return minElem\n}\n\n/**\n * Returns the first
element yielding the smallest value of the given function or `null` if there are no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.minBy\n * \n public inline fun <R : Comparable<R>>
CharArray.minBy(selector: (Char) -> R): Char? {\n if (isEmpty()) return null\n var minElem = this[0]\n val
lastIndex = this.lastIndex\n if (lastIndex == 0) return minElem\n var minValue = selector(minElem)\n for (i in
1..lastIndex) {\n val e = this[i]\n val v = selector(e)\n if (minValue > v) {\n minElem = e\n
minValue = v\n }\n }\n return minElem\n}\n\n/**\n * Returns the first element having the smallest value
according to the provided [comparator] or `null` if there are no elements.\n * \n public fun <T> Array<out
T>.minWith(comparator: Comparator<in T>): T? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in
1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n return
min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator] or
`null` if there are no elements.\n * \n public fun ByteArray.minWith(comparator: Comparator<in Byte>): Byte? {\n
if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if
(comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns the first element having the
smallest value according to the provided [comparator] or `null` if there are no elements.\n * \n public fun
ShortArray.minWith(comparator: Comparator<in Short>): Short? {\n if (isEmpty()) return null\n var min =
this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n
return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator]
or `null` if there are no elements.\n * \n public fun IntArray.minWith(comparator: Comparator<in Int>): Int? {\n
if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if
(comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns the first element having the
smallest value according to the provided [comparator] or `null` if there are no elements.\n * \n public fun
LongArray.minWith(comparator: Comparator<in Long>): Long? {\n if (isEmpty()) return null\n var min =
this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n
return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator]
or `null` if there are no elements.\n * \n public fun FloatArray.minWith(comparator: Comparator<in Float>): Float?
{\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if
(comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns the first element having the
smallest value according to the provided [comparator] or `null` if there are no elements.\n * \n public fun
DoubleArray.minWith(comparator: Comparator<in Double>): Double? {\n if (isEmpty()) return null\n var min =
this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n
return min\n}\n\n/**\n * Returns the first element having the smallest value according to the provided [comparator]
or `null` if there are no elements.\n * \n public fun BooleanArray.minWith(comparator: Comparator<in Boolean>):
Boolean? {\n if (isEmpty()) return null\n var min = this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n
if (comparator.compare(min, e) > 0) min = e\n }\n return min\n}\n\n/**\n * Returns the first element having the
smallest value according to the provided [comparator] or `null` if there are no elements.\n * \n public fun
CharArray.minWith(comparator: Comparator<in Char>): Char? {\n if (isEmpty()) return null\n var min =
this[0]\n for (i in 1..lastIndex) {\n val e = this[i]\n if (comparator.compare(min, e) > 0) min = e\n }\n
return min\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * \n * @sample
samples.collections.Collections.Aggregates.none\n * \n public fun <T> Array<out T>.none(): Boolean {\n return
isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * \n * @sample
samples.collections.Collections.Aggregates.none\n * \n public fun ByteArray.none(): Boolean {\n return
isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * \n * @sample

```

```

samples.collections.Collections.Aggregates.none\n *\npublic fun ShortArray.none(): Boolean {\n  return
isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.none\n *\npublic fun IntArray.none(): Boolean {\n  return
isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.none\n *\npublic fun LongArray.none(): Boolean {\n  return
isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.none\n *\npublic fun FloatArray.none(): Boolean {\n  return
isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.none\n *\npublic fun DoubleArray.none(): Boolean {\n  return
isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.none\n *\npublic fun BooleanArray.none(): Boolean {\n  return
isEmpty()\n}\n\n/**\n * Returns `true` if the array has no elements.\n * \n * @sample
samples.collections.Collections.Aggregates.none\n *\npublic fun CharArray.none(): Boolean {\n  return
isEmpty()\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n *\npublic inline fun <T> Array<out
T>.none(predicate: (T) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return false\n  return
true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n *\npublic inline fun ByteArray.none(predicate:
(Byte) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return false\n  return
true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n *\npublic inline fun ShortArray.none(predicate:
(Short) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return false\n  return
true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n *\npublic inline fun IntArray.none(predicate: (Int)
-> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return false\n  return true\n}\n\n/**\n *
Returns `true` if no elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n *\npublic inline fun LongArray.none(predicate:
(Long) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return false\n  return
true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n *\npublic inline fun FloatArray.none(predicate:
(Float) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return false\n  return
true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n *\npublic inline fun DoubleArray.none(predicate:
(Double) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return false\n  return
true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n *\npublic inline fun
BooleanArray.none(predicate: (Boolean) -> Boolean): Boolean {\n  for (element in this) if (predicate(element))
return false\n  return true\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n *\npublic inline fun CharArray.none(predicate:
(Char) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return false\n  return
true\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to
current accumulator value and each element.\n *\npublic inline fun <S, T : S> Array<out T>.reduce(operation:
(acc: S, T) -> S): S {\n  if (isEmpty())\n    throw UnsupportedOperationException("Empty array can't be
reduced.")\n  var accumulator: S = this[0]\n  for (index in 1..lastIndex) {\n    accumulator =
operation(accumulator, this[index])\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with the
first element and applying [operation] from left to right to current accumulator value and each element.\n *\npublic
inline fun ByteArray.reduce(operation: (acc: Byte, Byte) -> Byte): Byte {\n  if (isEmpty())\n    throw
UnsupportedOperationException("Empty array can't be reduced.")\n  var accumulator = this[0]\n  for (index in

```

```

1..lastIndex) {\n    accumulator = operation(accumulator, this[index])\n } \n return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.\n */\npublic inline fun ShortArray.reduce(operation: (acc: Short, Short) -> Short): Short {\n    if (isEmpty())\n        throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.\n */\npublic inline fun IntArray.reduce(operation: (acc: Int, Int) -> Int): Int {\n    if (isEmpty())\n        throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.\n */\npublic inline fun LongArray.reduce(operation: (acc: Long, Long) -> Long): Long {\n    if (isEmpty())\n        throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.\n */\npublic inline fun FloatArray.reduce(operation: (acc: Float, Float) -> Float): Float {\n    if (isEmpty())\n        throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.\n */\npublic inline fun DoubleArray.reduce(operation: (acc: Double, Double) -> Double): Double {\n    if (isEmpty())\n        throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.\n */\npublic inline fun BooleanArray.reduce(operation: (acc: Boolean, Boolean) -> Boolean): Boolean {\n    if (isEmpty())\n        throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.\n */\npublic inline fun CharArray.reduce(operation: (acc: Char, Char) -> Char): Char {\n    if (isEmpty())\n        throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.\n */\n@param [operation] function that takes the index of an element, current accumulator value\n * and the element itself and calculates the next accumulator value.\n */\npublic inline fun <S, T : S> Array<out T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {\n    if (isEmpty())\n        throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator: S = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(index, accumulator, this[index])\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.\n */\n@param [operation] function that takes the index of an element, current accumulator value\n * and the element itself and calculates the next accumulator value.\n */\npublic inline fun ByteArray.reduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): Byte {\n    if (isEmpty())\n        throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(index, accumulator, this[index])\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.\n */\n@param [operation] function that takes the index of an element, current accumulator value\n * and the element itself and calculates the next accumulator value.\n */\npublic inline fun

```

```

ShortArray.reduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): Short {
  if (isEmpty()) throw
  UnsupportedOperationException("Empty array can't be reduced.")
  var accumulator = this[0]
  for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
  }
  return accumulator
}
/** Accumulates value starting with the first element and applying [operation] from left to
right to current accumulator value and each element with its index in the original array.
@param [operation] function that takes the index of an element, current accumulator value
and the element itself and calculates the next accumulator value.
*/
public inline fun IntArray.reduceIndexed(operation: (index: Int, acc: Int, Int) -> Int):
Int {
  if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")
  var accumulator = this[0]
  for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator,
this[index])
  }
  return accumulator
}
/** Accumulates value starting with the first element and
applying [operation] from left to right to current accumulator value and each element with its index in the
original array.
@param [operation] function that takes the index of an element, current accumulator value
and the element itself and calculates the next accumulator value.
*/
public inline fun
LongArray.reduceIndexed(operation: (index: Int, acc: Long, Long) -> Long): Long {
  if (isEmpty()) throw
  UnsupportedOperationException("Empty array can't be reduced.")
  var accumulator = this[0]
  for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
  }
  return accumulator
}
/** Accumulates value starting with the first element and applying [operation] from left to
right to current accumulator value and each element with its index in the original array.
@param [operation] function that takes the index of an element, current accumulator value
and the element itself and calculates the next accumulator value.
*/
public inline fun FloatArray.reduceIndexed(operation: (index: Int, acc: Float, Float) -
> Float): Float {
  if (isEmpty()) throw UnsupportedOperationException("Empty array can't be
reduced.")
  var accumulator = this[0]
  for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
  }
  return accumulator
}
/** Accumulates value starting with the first
element and applying [operation] from left to right to current accumulator value and each element with its index
in the original array.
@param [operation] function that takes the index of an element, current accumulator
value and the element itself and calculates the next accumulator value.
*/
public inline fun
DoubleArray.reduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): Double {
  if (isEmpty())
  throw UnsupportedOperationException("Empty array can't be reduced.")
  var accumulator = this[0]
  for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
  }
  return accumulator
}
/** Accumulates value starting with the first element and applying [operation] from left to
right to current accumulator value and each element with its index in the original array.
@param [operation] function that takes the index of an element, current accumulator value
and the element itself and calculates the next accumulator value.
*/
public inline fun BooleanArray.reduceIndexed(operation: (index: Int, acc: Boolean,
Boolean) -> Boolean): Boolean {
  if (isEmpty()) throw UnsupportedOperationException("Empty array
can't be reduced.")
  var accumulator = this[0]
  for (index in 1..lastIndex) {
    accumulator =
operation(index, accumulator, this[index])
  }
  return accumulator
}
/** Accumulates value starting
with the first element and applying [operation] from left to right to current accumulator value and each element
with its index in the original array.
@param [operation] function that takes the index of an element, current
accumulator value and the element itself and calculates the next accumulator value.
*/
public inline fun
CharArray.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {
  if (isEmpty()) throw
  UnsupportedOperationException("Empty array can't be reduced.")
  var accumulator = this[0]
  for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
  }
  return accumulator
}
/** Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.
*/
public inline fun <S, T : S> Array<out
T>.reduceRight(operation: (T, acc: S) -> S): S {
  var index = lastIndex
  if (index < 0) throw
  UnsupportedOperationException("Empty array can't be reduced.")
  var accumulator: S = get(index--)
  while (index >= 0) {
    accumulator = operation(get(index--), accumulator)
  }
  return accumulator
}
/** Accumulates value starting with last element and applying [operation] from right to left to each element and current

```

```

accumulator value.\n */\npublic inline fun ByteArray.reduceRight(operation: (Byte, acc: Byte) -> Byte): Byte {\n
var index = lastIndex\n if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n
var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n
}\n return accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from
right to left to each element and current accumulator value.\n */\npublic inline fun
ShortArray.reduceRight(operation: (Short, acc: Short) -> Short): Short {\n var index = lastIndex\n if (index < 0)
throw UnsupportedOperationException("Empty array can't be reduced.")\n var accumulator = get(index--)\n
while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return
accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.\n */\npublic inline fun IntArray.reduceRight(operation: (Int, acc:
Int) -> Int): Int {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException("Empty array
can't be reduced.")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator =
operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with last
element and applying [operation] from right to left to each element and current accumulator value.\n */\npublic
inline fun LongArray.reduceRight(operation: (Long, acc: Long) -> Long): Long {\n var index = lastIndex\n if
(index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n var accumulator =
get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return
accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.\n */\npublic inline fun FloatArray.reduceRight(operation: (Float,
acc: Float) -> Float): Float {\n var index = lastIndex\n if (index < 0) throw
UnsupportedOperationException("Empty array can't be reduced.")\n var accumulator = get(index--)\n while
(index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n *
Accumulates value starting with last element and applying [operation] from right to left to each element and current
accumulator value.\n */\npublic inline fun DoubleArray.reduceRight(operation: (Double, acc: Double) -> Double):
Double {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException("Empty array can't be
reduced.")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--),
accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying
[operation] from right to left to each element and current accumulator value.\n */\npublic inline fun
BooleanArray.reduceRight(operation: (Boolean, acc: Boolean) -> Boolean): Boolean {\n var index = lastIndex\n
if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n var accumulator =
get(index--)\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n return
accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.\n */\npublic inline fun CharArray.reduceRight(operation: (Char, acc:
Char) -> Char): Char {\n var index = lastIndex\n if (index < 0) throw UnsupportedOperationException("Empty
array can't be reduced.")\n var accumulator = get(index--)\n while (index >= 0) {\n accumulator =
operation(get(index--), accumulator)\n }\n return accumulator\n}\n\n/**\n * Accumulates value starting with last
element and applying [operation] from right to left\n * to each element with its index in the original array and
current accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n *
and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <S, T : S>
Array<out T>.reduceRightIndexed(operation: (index: Int, T, acc: S) -> S): S {\n var index = lastIndex\n if (index
< 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n var accumulator: S = get(index--
)\n while (index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n
return accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right
to left\n * to each element with its index in the original array and current accumulator value.\n * @param
[operation] function that takes the index of an element, the element itself\n * and current accumulator value, and
calculates the next accumulator value.\n */\npublic inline fun ByteArray.reduceRightIndexed(operation: (index: Int,
Byte, acc: Byte) -> Byte): Byte {\n var index = lastIndex\n if (index < 0) throw
UnsupportedOperationException("Empty array can't be reduced.")\n var accumulator = get(index--)\n while

```

```

(index >= 0) {\n    accumulator = operation(index, get(index), accumulator)\n    --index\n } return
accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right to left\n * to each element with its index in the original array and current accumulator value.\n * @param [operation]
function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the
next accumulator value.\n */\npublic inline fun ShortArray.reduceRightIndexed(operation: (index: Int, Short, acc:
Short) -> Short): Short {\n    var index = lastIndex\n    if (index < 0) throw
UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = get(index--)\n    while
(index >= 0) {\n        accumulator = operation(index, get(index), accumulator)\n        --index\n    }\n    return
accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right to left\n *
to each element with its index in the original array and current accumulator value.\n * @param [operation]
function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the
next accumulator value.\n */\npublic inline fun IntArray.reduceRightIndexed(operation: (index: Int, Int, acc: Int) ->
Int): Int {\n    var index = lastIndex\n    if (index < 0) throw UnsupportedOperationException("Empty array can't be
reduced.")\n    var accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index,
get(index), accumulator)\n        --index\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with
last element and applying [operation] from right to left\n * to each element with its index in the original array and
current accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n *
and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun
LongArray.reduceRightIndexed(operation: (index: Int, Long, acc: Long) -> Long): Long {\n    var index =
lastIndex\n    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n    var
accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index),
accumulator)\n        --index\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with last element
and applying [operation] from right to left\n * to each element with its index in the original array and current
accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n * and
current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun
FloatArray.reduceRightIndexed(operation: (index: Int, Float, acc: Float) -> Float): Float {\n    var index =
lastIndex\n    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n    var
accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index),
accumulator)\n        --index\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with last element
and applying [operation] from right to left\n * to each element with its index in the original array and current
accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n * and
current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun
DoubleArray.reduceRightIndexed(operation: (index: Int, Double, acc: Double) -> Double): Double {\n    var index =
lastIndex\n    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n    var
accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index),
accumulator)\n        --index\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with last element
and applying [operation] from right to left\n * to each element with its index in the original array and current
accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n * and
current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun
BooleanArray.reduceRightIndexed(operation: (index: Int, Boolean, acc: Boolean) -> Boolean): Boolean {\n    var
index = lastIndex\n    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n    var
accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index),
accumulator)\n        --index\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with last element
and applying [operation] from right to left\n * to each element with its index in the original array and current
accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n * and
current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun
CharArray.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {\n    var index = lastIndex\n
if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator =

```

```

get(index--)\n while (index >= 0) {\n     accumulator = operation(index, get(index), accumulator)\n     --index\n }
\n return accumulator\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to
each element in the array.\n */\npublic inline fun <T> Array<out T>.sumBy(selector: (T) -> Int): Int {\n    var sum:
Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns the
sum of all values produced by [selector] function applied to each element in the array.\n */\npublic inline fun
ByteArray.sumBy(selector: (Byte) -> Int): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum +=
selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function
applied to each element in the array.\n */\npublic inline fun ShortArray.sumBy(selector: (Short) -> Int): Int {\n    var
sum: Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns
the sum of all values produced by [selector] function applied to each element in the array.\n */\npublic inline fun
IntArray.sumBy(selector: (Int) -> Int): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum +=
selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function
applied to each element in the array.\n */\npublic inline fun LongArray.sumBy(selector: (Long) -> Int): Int {\n    var
sum: Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns
the sum of all values produced by [selector] function applied to each element in the array.\n */\npublic inline fun
FloatArray.sumBy(selector: (Float) -> Int): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum +=
selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function
applied to each element in the array.\n */\npublic inline fun DoubleArray.sumBy(selector: (Double) -> Int): Int {\n
var sum: Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}\n\n/**\n *
Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\npublic
inline fun BooleanArray.sumBy(selector: (Boolean) -> Int): Int {\n    var sum: Int = 0\n    for (element in this) {\n
sum += selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector]
function applied to each element in the array.\n */\npublic inline fun CharArray.sumBy(selector: (Char) -> Int): Int
{\n    var sum: Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}\n\n/**\n *
Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\npublic
inline fun <T> Array<out T>.sumByDouble(selector: (T) -> Double): Double {\n    var sum: Double = 0.0\n    for
(element in this) {\n        sum += selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all values
produced by [selector] function applied to each element in the array.\n */\npublic inline fun
ByteArray.sumByDouble(selector: (Byte) -> Double): Double {\n    var sum: Double = 0.0\n    for (element in this)
{\n        sum += selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n */\npublic inline fun ShortArray.sumByDouble(selector:
(Short) -> Double): Double {\n    var sum: Double = 0.0\n    for (element in this) {\n        sum += selector(element)\n
}\n    return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each
element in the array.\n */\npublic inline fun IntArray.sumByDouble(selector: (Int) -> Double): Double {\n    var
sum: Double = 0.0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}\n\n/**\n *
Returns the sum of all values produced by [selector] function applied to each element in the array.\n */\npublic
inline fun LongArray.sumByDouble(selector: (Long) -> Double): Double {\n    var sum: Double = 0.0\n    for
(element in this) {\n        sum += selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all values
produced by [selector] function applied to each element in the array.\n */\npublic inline fun
FloatArray.sumByDouble(selector: (Float) -> Double): Double {\n    var sum: Double = 0.0\n    for (element in this)
{\n        sum += selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n */\npublic inline fun DoubleArray.sumByDouble(selector:
(Double) -> Double): Double {\n    var sum: Double = 0.0\n    for (element in this) {\n        sum +=
selector(element)\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function
applied to each element in the array.\n */\npublic inline fun BooleanArray.sumByDouble(selector: (Boolean) ->
Double): Double {\n    var sum: Double = 0.0\n    for (element in this) {\n        sum += selector(element)\n    }\n
return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in
the array.\n */\npublic inline fun CharArray.sumByDouble(selector: (Char) -> Double): Double {\n    var sum:

```

```

Double = 0.0\n for (element in this) {\n    sum += selector(element)\n } \n return sum\n}\n\n/**\n * Returns
an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are
any `null` elements.\n */\npublic fun <T : Any> Array<T?.>.requireNoNulls(): Array<T> {\n for (element in this)
{\n if (element == null) {\n throw IllegalArgumentException("`null element found in $this.`")\n } \n
}\n @Suppress(`UNCHECKED_CAST`)\n return this as Array<T>\n}\n\n/**\n * Splits the original array into
pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list
contains elements for which [predicate] yielded `false`.\n */\npublic inline fun <T> Array<out
T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {\n val first = ArrayList<T>()\n val second =
ArrayList<T>()\n for (element in this) {\n if (predicate(element)) {\n first.add(element)\n } else {\n
second.add(element)\n } \n } \n return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair
of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains
elements for which [predicate] yielded `false`.\n */\npublic inline fun ByteArray.partition(predicate: (Byte) ->
Boolean): Pair<List<Byte>, List<Byte>> {\n val first = ArrayList<Byte>()\n val second = ArrayList<Byte>()\n
for (element in this) {\n if (predicate(element)) {\n first.add(element)\n } else {\n
second.add(element)\n } \n } \n return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair of
lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains
elements for which [predicate] yielded `false`.\n */\npublic inline fun ShortArray.partition(predicate: (Short) ->
Boolean): Pair<List<Short>, List<Short>> {\n val first = ArrayList<Short>()\n val second =
ArrayList<Short>()\n for (element in this) {\n if (predicate(element)) {\n first.add(element)\n }
else {\n second.add(element)\n } \n } \n return Pair(first, second)\n}\n\n/**\n * Splits the original array
into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list
contains elements for which [predicate] yielded `false`.\n */\npublic inline fun IntArray.partition(predicate: (Int) ->
Boolean): Pair<List<Int>, List<Int>> {\n val first = ArrayList<Int>()\n val second = ArrayList<Int>()\n
for (element in this) {\n if (predicate(element)) {\n first.add(element)\n } else {\n
second.add(element)\n } \n } \n return Pair(first, second)\n}\n\n/**\n * Splits the original array into pair of
lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list contains
elements for which [predicate] yielded `false`.\n */\npublic inline fun LongArray.partition(predicate: (Long) ->
Boolean): Pair<List<Long>, List<Long>> {\n val first = ArrayList<Long>()\n val second =
ArrayList<Long>()\n for (element in this) {\n if (predicate(element)) {\n first.add(element)\n } else
{\n second.add(element)\n } \n } \n return Pair(first, second)\n}\n\n/**\n * Splits the original array into
pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list
contains elements for which [predicate] yielded `false`.\n */\npublic inline fun FloatArray.partition(predicate: (Float)
-> Boolean): Pair<List<Float>, List<Float>> {\n val first = ArrayList<Float>()\n val second =
ArrayList<Float>()\n for (element in this) {\n if (predicate(element)) {\n first.add(element)\n } else
{\n second.add(element)\n } \n } \n return Pair(first, second)\n}\n\n/**\n * Splits the original array into
pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list
contains elements for which [predicate] yielded `false`.\n */\npublic inline fun DoubleArray.partition(predicate:
(Double) -> Boolean): Pair<List<Double>, List<Double>> {\n val first = ArrayList<Double>()\n val second =
ArrayList<Double>()\n for (element in this) {\n if (predicate(element)) {\n first.add(element)\n }
else {\n second.add(element)\n } \n } \n return Pair(first, second)\n}\n\n/**\n * Splits the original array
into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while *second* list
contains elements for which [predicate] yielded `false`.\n */\npublic inline fun BooleanArray.partition(predicate:
(Boolean) -> Boolean): Pair<List<Boolean>, List<Boolean>> {\n val first = ArrayList<Boolean>()\n val second
= ArrayList<Boolean>()\n for (element in this) {\n if (predicate(element)) {\n first.add(element)\n
} else {\n second.add(element)\n } \n } \n return Pair(first, second)\n}\n\n/**\n * Splits the original
array into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n * while
*second* list contains elements for which [predicate] yielded `false`.\n */\npublic inline fun
CharArray.partition(predicate: (Char) -> Boolean): Pair<List<Char>, List<Char>> {\n val first =

```



```

ArrayList<Char>()\n    val second = ArrayList<Char>()\n    for (element in this) {\n        if (predicate(element)) {\n            first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return Pair(first, second)\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterable\n */\n\npublic infix fun <T, R> Array<out T>.zip(other: Array<out R>): List<Pair<T, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterable\n */\n\npublic infix fun <R> ByteArray.zip(other: Array<out R>): List<Pair<Byte, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterable\n */\n\npublic infix fun <R> ShortArray.zip(other: Array<out R>): List<Pair<Short, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterable\n */\n\npublic infix fun <R> IntArray.zip(other: Array<out R>): List<Pair<Int, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterable\n */\n\npublic infix fun <R> LongArray.zip(other: Array<out R>): List<Pair<Long, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterable\n */\n\npublic infix fun <R> FloatArray.zip(other: Array<out R>): List<Pair<Float, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterable\n */\n\npublic infix fun <R> DoubleArray.zip(other: Array<out R>): List<Pair<Double, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterable\n */\n\npublic infix fun <R> BooleanArray.zip(other: Array<out R>): List<Pair<Boolean, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterable\n */\n\npublic infix fun <R> CharArray.zip(other: Array<out R>): List<Pair<Char, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n\npublic inline fun <T, R, V> Array<out T>.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {\n    val size = minOf(size, other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n\npublic inline fun <R, V> ByteArray.zip(other: Array<out R>, transform: (a: Byte, b: R) -> V): List<V> {\n    val size = minOf(size, other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample\n samples.collections.Iterables.Operations.zipIterableWithTransform\n */\n\npublic inline fun <R, V>

```

```

ShortArray.zip(other: Array<out R>, transform: (a: Short, b: R) -> V): List<V> {\n  val size = minOf(size,
other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n  }\n  return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V>
IntArray.zip(other: Array<out R>, transform: (a: Int, b: R) -> V): List<V> {\n  val size = minOf(size, other.size)\n
  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n  }\n  return
list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same
index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length
of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n
*/\npublic inline fun <R, V> LongArray.zip(other: Array<out R>, transform: (a: Long, b: R) -> V): List<V> {\n
  val size = minOf(size, other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n
list.add(transform(this[i], other[i]))\n  }\n  return list\n}\n\n/**\n * Returns a list of values built from the elements
of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each
pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V>
FloatArray.zip(other: Array<out R>, transform: (a: Float, b: R) -> V): List<V> {\n  val size = minOf(size,
other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n
}\n  return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V>
DoubleArray.zip(other: Array<out R>, transform: (a: Double, b: R) -> V): List<V> {\n  val size = minOf(size,
other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n
}\n  return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V>
BooleanArray.zip(other: Array<out R>, transform: (a: Boolean, b: R) -> V): List<V> {\n  val size = minOf(size,
other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n
}\n  return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V>
CharArray.zip(other: Array<out R>, transform: (a: Char, b: R) -> V): List<V> {\n  val size = minOf(size,
other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n
}\n  return list\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with
the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <T, R> Array<out T>.zip(other:
Iterable<R>): List<Pair<T, R>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built
from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the
shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R>
ByteArray.zip(other: Iterable<R>): List<Pair<Byte, R>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n *
Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The
returned list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> ShortArray.zip(other: Iterable<R>):
List<Pair<Short, R>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the

```

elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> IntArray.zip(other: Iterable<R>): List<Pair<Int, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> LongArray.zip(other: Iterable<R>): List<Pair<Long, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> FloatArray.zip(other: Iterable<R>): List<Pair<Float, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> DoubleArray.zip(other: Iterable<R>): List<Pair<Double, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <R> BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic inline fun <T, R, V> Array<out T>.zip(other: Iterable<R>, transform: (a: T, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> ByteArray.zip(other: Iterable<R>, transform: (a: Byte, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> ShortArray.zip(other: Iterable<R>, transform: (a: Short, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <R, V> IntArray.zip(other: Iterable<R>, transform: (a: Int, b: R) -> V): List<V> {\n val arraySize = size\n val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n var i = 0\n for (element in other) {\n if (i >= arraySize) break\n list.add(transform(this[i++], element))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest

```

collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n * \npublic inline
fun <R, V> LongArray.zip(other: Iterable<R>, transform: (a: Long, b: R) -> V): List<V> {\n  val arraySize =
size\n  val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n  var i = 0\n  for (element
in other) {\n    if (i >= arraySize) break\n    list.add(transform(this[i++], element))\n  }\n  return
list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the
same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has
length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n * \npublic inline fun <R, V>
FloatArray.zip(other: Iterable<R>, transform: (a: Float, b: R) -> V): List<V> {\n  val arraySize = size\n  val list =
ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n  var i = 0\n  for (element in other) {\n
if (i >= arraySize) break\n  list.add(transform(this[i++], element))\n  }\n  return list\n}\n\n/**\n * Returns a
list of values built from the elements of `this` array and the [other] collection with the same index\n * using the
provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest
collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n * \npublic inline
fun <R, V> DoubleArray.zip(other: Iterable<R>, transform: (a: Double, b: R) -> V): List<V> {\n  val arraySize =
size\n  val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n  var i = 0\n  for (element
in other) {\n    if (i >= arraySize) break\n    list.add(transform(this[i++], element))\n  }\n  return
list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] collection with the
same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has
length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n * \npublic inline fun <R, V>
BooleanArray.zip(other: Iterable<R>, transform: (a: Boolean, b: R) -> V): List<V> {\n  val arraySize = size\n  val
list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n  var i = 0\n  for (element in other)
{\n    if (i >= arraySize) break\n    list.add(transform(this[i++], element))\n  }\n  return list\n}\n\n/**\n *
Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n *
using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the
shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n * \npublic
inline fun <R, V> CharArray.zip(other: Iterable<R>, transform: (a: Char, b: R) -> V): List<V> {\n  val
arraySize = size\n  val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n  var i = 0\n
for (element in other) {\n    if (i >= arraySize) break\n    list.add(transform(this[i++], element))\n  }\n  return
list\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same
index.\n * The returned list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterable\n * \npublic infix fun ByteArray.zip(other: ByteArray):
List<Pair<Byte, Byte>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from
the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the
shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n * \npublic infix fun
ShortArray.zip(other: ShortArray): List<Pair<Short, Short>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n
* Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The
returned list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterable\n * \npublic infix fun IntArray.zip(other: IntArray):
List<Pair<Int, Int>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from
the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the
shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n * \npublic infix fun
LongArray.zip(other: LongArray): List<Pair<Long, Long>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n
* Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The
returned list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterable\n * \npublic infix fun FloatArray.zip(other: FloatArray):
List<Pair<Float, Float>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from

```

the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun DoubleArray.zip(other: DoubleArray): List<Pair<Double, Double>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun BooleanArray.zip(other: BooleanArray): List<Pair<Boolean, Boolean>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun CharArray.zip(other: CharArray): List<Pair<Char, Char>> {\n return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> ByteArray.zip(other: ByteArray, transform: (a: Byte, b: Byte) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> ShortArray.zip(other: ShortArray, transform: (a: Short, b: Short) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> IntArray.zip(other: IntArray, transform: (a: Int, b: Int) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> LongArray.zip(other: LongArray, transform: (a: Long, b: Long) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> FloatArray.zip(other: FloatArray, transform: (a: Float, b: Float) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V> DoubleArray.zip(other: DoubleArray, transform: (a: Double, b: Double) -> V): List<V> {\n val size = minOf(size, other.size)\n val list = ArrayList<V>(size)\n for (i in 0 until size) {\n list.add(transform(this[i], other[i]))\n }\n return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V>

```

BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> {\n  val size =
minOf(size, other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i],
other[i]))\n  }\n  return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the
[other] array with the same index\n * using the provided [transform] function applied to each pair of elements.\n *
The returned list has length of the shortest array.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <V>
CharArray.zip(other: CharArray, transform: (a: Char, b: Char) -> V): List<V> {\n  val size = minOf(size,
other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n
}\n  return list\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the
given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value
of [limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string
(which defaults to "...").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun
<T, A : Appendable> Array<out T>.joinTo(buffer: A, separator: CharSequence = "\", \"", prefix: CharSequence = "\"",
postfix: CharSequence = "\"", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? =
null): A {\n  buffer.append(prefix)\n  var count = 0\n  for (element in this) {\n    if (++count > 1)
buffer.append(separator)\n    if (limit < 0 || count <= limit) {\n      buffer.appendElement(element, transform)\n
    } else break\n  }\n  if (limit >= 0 && count > limit) buffer.append(truncated)\n  buffer.append(postfix)\n
return buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which
defaults to "...").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A :
Appendable> ByteArray.joinTo(buffer: A, separator: CharSequence = "\", \"", prefix: CharSequence = "\"", postfix:
CharSequence = "\"", limit: Int = -1, truncated: CharSequence = "...", transform: ((Byte) -> CharSequence)? =
null): A {\n  buffer.append(prefix)\n  var count = 0\n  for (element in this) {\n    if (++count > 1)
buffer.append(separator)\n    if (limit < 0 || count <= limit) {\n      if (transform != null)\n        buffer.append(transform(element))\n      else\n        buffer.append(element.toString())\n    } else break\n
}\n  if (limit >= 0 && count > limit) buffer.append(truncated)\n  buffer.append(postfix)\n  return
buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which
defaults to "...").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A :
Appendable> ShortArray.joinTo(buffer: A, separator: CharSequence = "\", \"", prefix: CharSequence = "\"", postfix:
CharSequence = "\"", limit: Int = -1, truncated: CharSequence = "...", transform: ((Short) -> CharSequence)? =
null): A {\n  buffer.append(prefix)\n  var count = 0\n  for (element in this) {\n    if (++count > 1)
buffer.append(separator)\n    if (limit < 0 || count <= limit) {\n      if (transform != null)\n        buffer.append(transform(element))\n      else\n        buffer.append(element.toString())\n    } else break\n
}\n  if (limit >= 0 && count > limit) buffer.append(truncated)\n  buffer.append(postfix)\n  return
buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which
defaults to "...").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n */\npublic fun <A :
Appendable> IntArray.joinTo(buffer: A, separator: CharSequence = "\", \"", prefix: CharSequence = "\"", postfix:
CharSequence = "\"", limit: Int = -1, truncated: CharSequence = "...", transform: ((Int) -> CharSequence)? =
null): A {\n  buffer.append(prefix)\n  var count = 0\n  for (element in this) {\n    if (++count > 1)
buffer.append(separator)\n    if (limit < 0 || count <= limit) {\n      if (transform != null)\n        buffer.append(transform(element))\n      else\n        buffer.append(element.toString())\n    } else break\n
}\n  if (limit >= 0 && count > limit) buffer.append(truncated)\n  buffer.append(postfix)\n  return
buffer\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and using the given

```

```
[prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of  
[limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which  
defaults to "...").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n * \npublic fun <A :  
Appendable> LongArray.joinTo(buffer: A, separator: CharSequence = "|", prefix: CharSequence = "", postfix:  
CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Long) -> CharSequence)? =  
null): A {\n  buffer.append(prefix)\n  var count = 0\n  for (element in this) {\n    if (++count > 1)  
buffer.append(separator)\n    if (limit < 0 || count <= limit) {\n      if (transform != null)\nbuffer.append(transform(element))\n      else\n        buffer.append(element.toString())\n    } else break\n  }\n  if (limit >= 0 && count > limit) buffer.append(truncated)\n  buffer.append(postfix)\n  return  
buffer}\n\n\n * Appends the string from all the elements separated using [separator] and using the given  
[prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of  
[limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which  
defaults to "...").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n * \npublic fun <A :  
Appendable> FloatArray.joinTo(buffer: A, separator: CharSequence = "|", prefix: CharSequence = "", postfix:  
CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Float) -> CharSequence)? =  
null): A {\n  buffer.append(prefix)\n  var count = 0\n  for (element in this) {\n    if (++count > 1)  
buffer.append(separator)\n    if (limit < 0 || count <= limit) {\n      if (transform != null)\nbuffer.append(transform(element))\n      else\n        buffer.append(element.toString())\n    } else break\n  }\n  if (limit >= 0 && count > limit) buffer.append(truncated)\n  buffer.append(postfix)\n  return  
buffer}\n\n\n * Appends the string from all the elements separated using [separator] and using the given  
[prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of  
[limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which  
defaults to "...").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n * \npublic fun <A :  
Appendable> DoubleArray.joinTo(buffer: A, separator: CharSequence = "|", prefix: CharSequence = "", postfix:  
CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Double) -> CharSequence)? =  
null): A {\n  buffer.append(prefix)\n  var count = 0\n  for (element in this) {\n    if (++count > 1)  
buffer.append(separator)\n    if (limit < 0 || count <= limit) {\n      if (transform != null)\nbuffer.append(transform(element))\n      else\n        buffer.append(element.toString())\n    } else break\n  }\n  if (limit >= 0 && count > limit) buffer.append(truncated)\n  buffer.append(postfix)\n  return  
buffer}\n\n\n * Appends the string from all the elements separated using [separator] and using the given  
[prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of  
[limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which  
defaults to "...").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n * \npublic fun <A :  
Appendable> BooleanArray.joinTo(buffer: A, separator: CharSequence = "|", prefix: CharSequence = "", postfix:  
CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Boolean) ->  
CharSequence)? = null): A {\n  buffer.append(prefix)\n  var count = 0\n  for (element in this) {\n    if (++count  
> 1) buffer.append(separator)\n    if (limit < 0 || count <= limit) {\n      if (transform != null)\nbuffer.append(transform(element))\n      else\n        buffer.append(element.toString())\n    } else break\n  }\n  if (limit >= 0 && count > limit) buffer.append(truncated)\n  buffer.append(postfix)\n  return  
buffer}\n\n\n * Appends the string from all the elements separated using [separator] and using the given  
[prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of  
[limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which  
defaults to "...").\n * \n * @sample samples.collections.Collections.Transformations.joinTo\n * \npublic fun <A :  
Appendable> CharArray.joinTo(buffer: A, separator: CharSequence = "|", prefix: CharSequence = "", postfix:  
CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Char) -> CharSequence)? =  
null): A {\n  buffer.append(prefix)\n  var count = 0\n  for (element in this) {\n    if (++count > 1)  
buffer.append(separator)\n    if (limit < 0 || count <= limit) {\n      if (transform != null)\nbuffer.append(transform(element))\n      else\n        buffer.append(element)\n    } else break\n  }\n  if
```

```

(limit >= 0 && count > limit) buffer.append(truncated)\n  buffer.append(postfix)\n  return buffer\n}\n\n/**\n *
Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if
supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only
the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to "...").\n * \n *
@sample samples.collections.Collections.Transformations.joinToString\n *\npublic fun <T> Array<out
T>.joinToString(separator: CharSequence = "\", \", prefix: CharSequence = "\", postfix: CharSequence = "\", limit:
Int = -1, truncated: CharSequence = "...\", transform: ((T) -> CharSequence)? = null): String {\n  return
joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a
string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n
* If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first
[limit]\n * elements will be appended, followed by the [truncated] string (which defaults to "...").\n * \n *
@sample
samples.collections.Collections.Transformations.joinToString\n *\npublic fun ByteArray.joinToString(separator:
CharSequence = "\", \", prefix: CharSequence = "\", postfix: CharSequence = "\", limit: Int = -1, truncated:
CharSequence = "...\", transform: ((Byte) -> CharSequence)? = null): String {\n  return joinTo(StringBuilder(),
separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements
separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be
huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be
appended, followed by the [truncated] string (which defaults to "...").\n * \n * @sample
samples.collections.Collections.Transformations.joinToString\n *\npublic fun ShortArray.joinToString(separator:
CharSequence = "\", \", prefix: CharSequence = "\", postfix: CharSequence = "\", limit: Int = -1, truncated:
CharSequence = "...\", transform: ((Short) -> CharSequence)? = null): String {\n  return joinTo(StringBuilder(),
separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements
separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be
huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be
appended, followed by the [truncated] string (which defaults to "...").\n * \n * @sample
samples.collections.Collections.Transformations.joinToString\n *\npublic fun IntArray.joinToString(separator:
CharSequence = "\", \", prefix: CharSequence = "\", postfix: CharSequence = "\", limit: Int = -1, truncated:
CharSequence = "...\", transform: ((Int) -> CharSequence)? = null): String {\n  return joinTo(StringBuilder(),
separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements
separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be
huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be
appended, followed by the [truncated] string (which defaults to "...").\n * \n * @sample
samples.collections.Collections.Transformations.joinToString\n *\npublic fun LongArray.joinToString(separator:
CharSequence = "\", \", prefix: CharSequence = "\", postfix: CharSequence = "\", limit: Int = -1, truncated:
CharSequence = "...\", transform: ((Long) -> CharSequence)? = null): String {\n  return joinTo(StringBuilder(),
separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements
separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be
huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be
appended, followed by the [truncated] string (which defaults to "...").\n * \n * @sample
samples.collections.Collections.Transformations.joinToString\n *\npublic fun FloatArray.joinToString(separator:
CharSequence = "\", \", prefix: CharSequence = "\", postfix: CharSequence = "\", limit: Int = -1, truncated:
CharSequence = "...\", transform: ((Float) -> CharSequence)? = null): String {\n  return joinTo(StringBuilder(),
separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates a string from all the elements
separated using [separator] and using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be
huge, you can specify a non-negative value of [limit], in which case only the first [limit]\n * elements will be
appended, followed by the [truncated] string (which defaults to "...").\n * \n * @sample
samples.collections.Collections.Transformations.joinToString\n *\npublic fun DoubleArray.joinToString(separator:
CharSequence = "\", \", prefix: CharSequence = "\", postfix: CharSequence = "\", limit: Int = -1, truncated:

```



```

CharSequence = "...\", transform: ((Double) -> CharSequence)? = null): String {
return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}
Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
@sample samples.collections.Collections.Transformations.joinToString
public fun BooleanArray.joinToString(separator: CharSequence = "\", \"\", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = "...\", transform: ((Boolean) -> CharSequence)? = null): String {
return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}
Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
@sample samples.collections.Collections.Transformations.joinToString
public fun CharArray.joinToString(separator: CharSequence = "\", \"\", prefix: CharSequence = \"\", postfix: CharSequence = \"\", limit: Int = -1, truncated: CharSequence = "...\", transform: ((Char) -> CharSequence)? = null): String {
return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}
Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
public fun <T> Array<out T>.asIterable(): Iterable<T> {
if (isEmpty()) return emptyList()
return Iterable { this.iterator() }}
Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
public fun ByteArray.asIterable(): Iterable<Byte> {
if (isEmpty()) return emptyList()
return Iterable { this.iterator() }}
Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
public fun ShortArray.asIterable(): Iterable<Short> {
if (isEmpty()) return emptyList()
return Iterable { this.iterator() }}
Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
public fun IntArray.asIterable(): Iterable<Int> {
if (isEmpty()) return emptyList()
return Iterable { this.iterator() }}
Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
public fun LongArray.asIterable(): Iterable<Long> {
if (isEmpty()) return emptyList()
return Iterable { this.iterator() }}
Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
public fun FloatArray.asIterable(): Iterable<Float> {
if (isEmpty()) return emptyList()
return Iterable { this.iterator() }}
Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
public fun DoubleArray.asIterable(): Iterable<Double> {
if (isEmpty()) return emptyList()
return Iterable { this.iterator() }}
Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
public fun BooleanArray.asIterable(): Iterable<Boolean> {
if (isEmpty()) return emptyList()
return Iterable { this.iterator() }}
Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
public fun CharArray.asIterable(): Iterable<Char> {
if (isEmpty()) return emptyList()
return Iterable { this.iterator() }}
Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
@sample samples.collections.Sequences.Building.sequenceFromArray
public fun <T> Array<out T>.asSequence(): Sequence<T> {
if (isEmpty()) return emptySequence()
return Sequence { this.iterator() }}
Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
@sample samples.collections.Sequences.Building.sequenceFromArray
public fun ByteArray.asSequence(): Sequence<Byte> {
if (isEmpty()) return emptySequence()
return Sequence { this.iterator() }}
Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
@sample samples.collections.Sequences.Building.sequenceFromArray
public fun ShortArray.asSequence(): Sequence<Short> {
if (isEmpty()) return emptySequence()
return Sequence { this.iterator() }}
Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
@sample samples.collections.Sequences.Building.sequenceFromArray
public fun IntArray.asSequence(): Sequence<Int> {
if (isEmpty()) return emptySequence()
return Sequence { this.iterator() }}

```

```

Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n *
@sample samples.collections.Sequences.Building.sequenceFromArray\n *^npublic fun LongArray.asSequence():
Sequence<Long> {\n if (isEmpty()) return emptySequence()\n return Sequence { this.iterator() }\n}\n\n/**\n *
Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n *
@sample samples.collections.Sequences.Building.sequenceFromArray\n *^npublic fun FloatArray.asSequence():
Sequence<Float> {\n if (isEmpty()) return emptySequence()\n return Sequence { this.iterator() }\n}\n\n/**\n *
Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n *
@sample samples.collections.Sequences.Building.sequenceFromArray\n *^npublic fun DoubleArray.asSequence():
Sequence<Double> {\n if (isEmpty()) return emptySequence()\n return Sequence { this.iterator() }\n}\n\n/**\n *
Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.\n * \n *
@sample samples.collections.Sequences.Building.sequenceFromArray\n *^npublic fun
BooleanArray.asSequence(): Sequence<Boolean> {\n if (isEmpty()) return emptySequence()\n return Sequence
{ this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original array returning its elements
when being iterated.\n * \n * @sample samples.collections.Sequences.Building.sequenceFromArray\n *^npublic
fun CharArray.asSequence(): Sequence<Char> {\n if (isEmpty()) return emptySequence()\n return Sequence {
this.iterator() }\n}\n\n/**\n * Returns an average value of elements in the array.\n
*\n@kotlin.jvm.JvmName("averageOfByte")\npublic fun Array<out Byte>.average(): Double {\n var sum:
Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n ++count\n }\n return
if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n
*\n@kotlin.jvm.JvmName("averageOfShort")\npublic fun Array<out Short>.average(): Double {\n var sum:
Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n ++count\n }\n return
if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n
*\n@kotlin.jvm.JvmName("averageOfInt")\npublic fun Array<out Int>.average(): Double {\n var sum: Double
= 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n ++count\n }\n return if (count
== 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n
*\n@kotlin.jvm.JvmName("averageOfLong")\npublic fun Array<out Long>.average(): Double {\n var sum:
Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n ++count\n }\n return
if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n
*\n@kotlin.jvm.JvmName("averageOfFloat")\npublic fun Array<out Float>.average(): Double {\n var sum:
Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n ++count\n }\n return
if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n
*\n@kotlin.jvm.JvmName("averageOfDouble")\npublic fun Array<out Double>.average(): Double {\n var sum:
Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n ++count\n }\n return
if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n
*\npublic fun ByteArray.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in
this) {\n sum += element\n ++count\n }\n return if (count == 0) Double.NaN else sum /
count\n}\n\n/**\n * Returns an average value of elements in the array.\n *\npublic fun ShortArray.average():
Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n
++count\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of
elements in the array.\n *\npublic fun IntArray.average(): Double {\n var sum: Double = 0.0\n var count: Int =
0\n for (element in this) {\n sum += element\n ++count\n }\n return if (count == 0) Double.NaN else
sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n *\npublic fun LongArray.average():
Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n sum += element\n
++count\n }\n return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of
elements in the array.\n *\npublic fun FloatArray.average(): Double {\n var sum: Double = 0.0\n var count: Int
= 0\n for (element in this) {\n sum += element\n ++count\n }\n return if (count == 0) Double.NaN
else sum / count\n}\n\n/**\n * Returns an average value of elements in the array.\n *\npublic fun
DoubleArray.average(): Double {\n var sum: Double = 0.0\n var count: Int = 0\n for (element in this) {\n

```

```

sum += element\n    ++count\n    }\n    return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns
the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName("sumOfByte")\npublic fun Array<out
Byte>.sum(): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum += element\n    }\n    return
sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName("sumOfShort")\npublic fun Array<out Short>.sum(): Int {\n    var sum: Int = 0\n    for
(element in this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all elements in the
array.\n */\n@kotlin.jvm.JvmName("sumOfInt")\npublic fun Array<out Int>.sum(): Int {\n    var sum: Int = 0\n
for (element in this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all elements in
the array.\n */\n@kotlin.jvm.JvmName("sumOfLong")\npublic fun Array<out Long>.sum(): Long {\n    var sum:
Long = 0L\n    for (element in this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n * Returns the sum of
all elements in the array.\n */\n@kotlin.jvm.JvmName("sumOfFloat")\npublic fun Array<out Float>.sum(): Float
{\n    var sum: Float = 0.0f\n    for (element in this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n *
Returns the sum of all elements in the array.\n */\n@kotlin.jvm.JvmName("sumOfDouble")\npublic fun Array<out
Double>.sum(): Double {\n    var sum: Double = 0.0\n    for (element in this) {\n        sum += element\n    }\n
return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\npublic fun ByteArray.sum(): Int {\n
var sum: Int = 0\n    for (element in this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n * Returns the
sum of all elements in the array.\n */\npublic fun ShortArray.sum(): Int {\n    var sum: Int = 0\n    for (element in
this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\npublic fun IntArray.sum(): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum += element\n    }\n
return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n */\npublic fun LongArray.sum(): Long {\n
var sum: Long = 0L\n    for (element in this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n * Returns the
sum of all elements in the array.\n */\npublic fun FloatArray.sum(): Float {\n    var sum: Float = 0.0f\n    for
(element in this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all elements in the
array.\n */\npublic fun DoubleArray.sum(): Double {\n    var sum: Double = 0.0\n    for (element in this) {\n
sum += element\n    }\n    return sum\n}\n\n"/**\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming
Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\n */\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("RangesKt")\n\npackage
kotlin.ranges\n\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n\nimport kotlin.random.*\n\n/**\n * Returns a
random element from this range.\n * \n * @throws IllegalArgumentException if this range is empty.\n */\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun IntRange.random(): Int {\n    return
random(Random)\n}\n\n/**\n * Returns a random element from this range.\n * \n * @throws
IllegalArgumentException if this range is empty.\n */\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic
inline fun LongRange.random(): Long {\n    return random(Random)\n}\n\n/**\n * Returns a random element from
this range.\n * \n * @throws IllegalArgumentException if this range is empty.\n */\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun CharRange.random(): Char {\n    return
random(Random)\n}\n\n/**\n * Returns a random element from this range using the specified source of
randomness.\n * \n * @throws IllegalArgumentException if this range is empty.\n */\n@SinceKotlin("1.3")\npublic fun IntRange.random(random: Random): Int {\n    try {\n        return
random.nextInt(this)\n    } catch (e: IllegalArgumentException) {\n        throw
NoSuchElementException(e.message)\n    }\n}\n\n/**\n * Returns a random element from this range using the
specified source of randomness.\n * \n * @throws IllegalArgumentException if this range is empty.\n */\n@SinceKotlin("1.3")\npublic fun LongRange.random(random: Random): Long {\n    try {\n        return
random.nextLong(this)\n    } catch (e: IllegalArgumentException) {\n        throw
NoSuchElementException(e.message)\n    }\n}\n\n/**\n * Returns a random element from this range using the
specified source of randomness.\n * \n * @throws IllegalArgumentException if this range is empty.\n */\n@SinceKotlin("1.3")\npublic fun CharRange.random(random: Random): Char {\n    try {\n        return

```

```

random.nextInt(first.toInt(), last.toInt() + 1).toChar()\n } catch(e: IllegalArgumentException) {\n     throw
NoSuchElementException(e.message)\n } }\n\n/**\n * Returns `true` if this range contains the specified
[element].\n * \n * Always returns `false` if the [element] is `null`.\n
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline operator fun IntRange.contains(element:
Int?): Boolean {\n     return element != null && contains(element)\n}\n\n/**\n * Returns `true` if this range contains
the specified [element].\n * \n * Always returns `false` if the [element] is `null`.\n
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline operator fun LongRange.contains(element:
Long?): Boolean {\n     return element != null && contains(element)\n}\n\n/**\n * Returns `true` if this range
contains the specified [element].\n * \n * Always returns `false` if the [element] is `null`.\n
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline operator fun CharRange.contains(element:
Char?): Boolean {\n     return element != null && contains(element)\n}\n\n/**\n * Checks if the specified [value]
belongs to this range.\n *\n@kotlin.jvm.JvmName("intRangeContains")\npublic operator fun
ClosedRange<Int>.contains(value: Byte): Boolean {\n     return contains(value.toInt())\n}\n\n/**\n * Checks if the
specified [value] belongs to this range.\n *\n@kotlin.jvm.JvmName("longRangeContains")\npublic operator fun
ClosedRange<Long>.contains(value: Byte): Boolean {\n     return contains(value.toLong())\n}\n\n/**\n * Checks if
the specified [value] belongs to this range.\n *\n@kotlin.jvm.JvmName("shortRangeContains")\npublic operator
fun ClosedRange<Short>.contains(value: Byte): Boolean {\n     return contains(value.toShort())\n}\n\n/**\n *
Checks if the specified [value] belongs to this range.\n *\n@Deprecated("This `contains` operation mixing integer
and floating point arguments has ambiguous semantics and is going to be
removed.")\n@kotlin.jvm.JvmName("doubleRangeContains")\npublic operator fun
ClosedRange<Double>.contains(value: Byte): Boolean {\n     return contains(value.toDouble())\n}\n\n/**\n *
Checks if the specified [value] belongs to this range.\n *\n@Deprecated("This `contains` operation mixing integer
and floating point arguments has ambiguous semantics and is going to be
removed.")\n@kotlin.jvm.JvmName("floatRangeContains")\npublic operator fun
ClosedRange<Float>.contains(value: Byte): Boolean {\n     return contains(value.toFloat())\n}\n\n/**\n * Checks if
the specified [value] belongs to this range.\n *\n@Deprecated("This `contains` operation mixing integer and
floating point arguments has ambiguous semantics and is going to be
removed.")\n@kotlin.jvm.JvmName("intRangeContains")\npublic operator fun ClosedRange<Int>.contains(value:
Double): Boolean {\n     return value.toIntExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n *
Checks if the specified [value] belongs to this range.\n *\n@Deprecated("This `contains` operation mixing integer
and floating point arguments has ambiguous semantics and is going to be
removed.")\n@kotlin.jvm.JvmName("longRangeContains")\npublic operator fun
ClosedRange<Long>.contains(value: Double): Boolean {\n     return value.toLongExactOrNull().let { if (it != null)
contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n
*\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics
and is going to be removed.")\n@kotlin.jvm.JvmName("byteRangeContains")\npublic operator fun
ClosedRange<Byte>.contains(value: Double): Boolean {\n     return value.toByteExactOrNull().let { if (it != null)
contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n
*\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics
and is going to be removed.")\n@kotlin.jvm.JvmName("shortRangeContains")\npublic operator fun
ClosedRange<Short>.contains(value: Double): Boolean {\n     return value.toShortExactOrNull().let { if (it != null)
contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n
*\n@kotlin.jvm.JvmName("floatRangeContains")\npublic operator fun ClosedRange<Float>.contains(value:
Double): Boolean {\n     return contains(value.toFloat())\n}\n\n/**\n * Checks if the specified [value] belongs to this
range.\n *\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous
semantics and is going to be removed.")\n@kotlin.jvm.JvmName("intRangeContains")\npublic operator fun
ClosedRange<Int>.contains(value: Float): Boolean {\n     return value.toIntExactOrNull().let { if (it != null)
contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

```

*\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")\n@kotlin.jvm.JvmName("longRangeContains")\npublic operator fun ClosedRange<Long>.contains(value: Float): Boolean {\n return value.toLongExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")\n@kotlin.jvm.JvmName("byteRangeContains")\npublic operator fun ClosedRange<Byte>.contains(value: Float): Boolean {\n return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")\n@kotlin.jvm.JvmName("shortRangeContains")\npublic operator fun ClosedRange<Short>.contains(value: Float): Boolean {\n return value.toShortExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@kotlin.jvm.JvmName("doubleRangeContains")\npublic operator fun ClosedRange<Double>.contains(value: Float): Boolean {\n return contains(value.toDouble())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@kotlin.jvm.JvmName("longRangeContains")\npublic operator fun ClosedRange<Long>.contains(value: Int): Boolean {\n return contains(value.toLong())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@kotlin.jvm.JvmName("byteRangeContains")\npublic operator fun ClosedRange<Byte>.contains(value: Int): Boolean {\n return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@kotlin.jvm.JvmName("shortRangeContains")\npublic operator fun ClosedRange<Short>.contains(value: Int): Boolean {\n return value.toShortExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")\n@kotlin.jvm.JvmName("doubleRangeContains")\npublic operator fun ClosedRange<Double>.contains(value: Int): Boolean {\n return contains(value.toDouble())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")\n@kotlin.jvm.JvmName("floatRangeContains")\npublic operator fun ClosedRange<Float>.contains(value: Int): Boolean {\n return contains(value.toFloat())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@kotlin.jvm.JvmName("intRangeContains")\npublic operator fun ClosedRange<Int>.contains(value: Long): Boolean {\n return value.toIntExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@kotlin.jvm.JvmName("byteRangeContains")\npublic operator fun ClosedRange<Byte>.contains(value: Long): Boolean {\n return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@kotlin.jvm.JvmName("shortRangeContains")\npublic operator fun ClosedRange<Short>.contains(value: Long): Boolean {\n return value.toShortExactOrNull().let { if (it != null) contains(it) else false }\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")\n@kotlin.jvm.JvmName("doubleRangeContains")\npublic operator fun ClosedRange<Double>.contains(value: Long): Boolean {\n return contains(value.toDouble())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")\n@kotlin.jvm.JvmName("floatRangeContains")\npublic operator fun ClosedRange<Float>.contains(value: Long): Boolean {\n return contains(value.toFloat())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@kotlin.jvm.JvmName("intRangeContains")\npublic operator fun ClosedRange<Int>.contains(value: Short): Boolean {\n return contains(value.toInt())\n}\n\n/**\n * Checks if the specified [value] belongs to this range.\n

*\n@kotlin.jvm.JvmName("longRangeContains")\npublic operator fun ClosedRange<Long>.contains(value: Short): Boolean {\n return contains(value.toLong())\n}\n\n/**\n * Checks if

the specified [value] belongs to this range.

```

@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Short): Boolean {
    return value.toByteExactOrNull().let { if (it != null) contains(it) else false }
}

```

* Checks if the specified [value] belongs to this range.

@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")

```

@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Short): Boolean {
    return contains(value.toDouble())
}

```

* Checks if the specified [value] belongs to this range.

@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")

```

@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Short): Boolean {
    return contains(value.toFloat())
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Int.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(this, to.toInt(), -1)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Long.downTo(to: Byte): LongProgression {
    return LongProgression.fromClosedRange(this, to.toLong(), -1L)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Byte.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Short.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Char.downTo(to: Char): CharProgression {
    return CharProgression.fromClosedRange(this, to, -1)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Int.downTo(to: Int): IntProgression {
    return IntProgression.fromClosedRange(this, to, -1)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Long.downTo(to: Int): LongProgression {
    return LongProgression.fromClosedRange(this, to.toLong(), -1L)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Byte.downTo(to: Int): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to, -1)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Short.downTo(to: Int): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to, -1)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Int.downTo(to: Long): LongProgression {
    return LongProgression.fromClosedRange(this.toLong(), to, -1L)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

```

public infix fun Long.downTo(to: Long): LongProgression {
    return LongProgression.fromClosedRange(this, to, -1L)
}

```

* Returns a progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Byte.downTo(to: Long): LongProgression {\n return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * \n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Short.downTo(to: Long): LongProgression {\n return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * \n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Int.downTo(to: Short): IntProgression {\n return IntProgression.fromClosedRange(this, to.toInt(), -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * \n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Long.downTo(to: Short): LongProgression {\n return LongProgression.fromClosedRange(this, to.toLong(), -1L)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * \n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Byte.downTo(to: Short): IntProgression {\n return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n}\n\n/**\n * Returns a progression from this value down to the specified [to] value with the step -1.\n * \n * The [to] value should be less than or equal to `this` value.\n * \n * If the [to] value is greater than `this` value the returned progression is empty.\n */\npublic infix fun Short.downTo(to: Short): IntProgression {\n return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n}\n\n/**\n * Returns a progression that goes over the same range in the opposite direction with the same step.\n */\npublic fun IntProgression.reversed(): IntProgression {\n return IntProgression.fromClosedRange(last, first, -step)\n}\n\n/**\n * Returns a progression that goes over the same range in the opposite direction with the same step.\n */\npublic fun LongProgression.reversed(): LongProgression {\n return LongProgression.fromClosedRange(last, first, -step)\n}\n\n/**\n * Returns a progression that goes over the same range in the opposite direction with the same step.\n */\npublic fun CharProgression.reversed(): CharProgression {\n return CharProgression.fromClosedRange(last, first, -step)\n}\n\n/**\n * Returns a progression that goes over the same range with the given step.\n */\npublic infix fun IntProgression.step(step: Int): IntProgression {\n checkStepIsPositive(step > 0, step)\n return IntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}\n\n/**\n * Returns a progression that goes over the same range with the given step.\n */\npublic infix fun LongProgression.step(step: Long): LongProgression {\n checkStepIsPositive(step > 0, step)\n return LongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}\n\n/**\n * Returns a progression that goes over the same range with the given step.\n */\npublic infix fun CharProgression.step(step: Int): CharProgression {\n checkStepIsPositive(step > 0, step)\n return CharProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}\n\ninternal fun Int.toByteExactOrNull(): Byte? {\n return if (this in Byte.MIN_VALUE.toInt()..Byte.MAX_VALUE.toInt()) this.toByte() else null\n}\n\ninternal fun Long.toByteExactOrNull(): Byte? {\n return if (this in Byte.MIN_VALUE.toLong()..Byte.MAX_VALUE.toLong()) this.toByte() else null\n}\n\ninternal fun Short.toByteExactOrNull(): Byte? {\n return if (this in Byte.MIN_VALUE.toShort()..Byte.MAX_VALUE.toShort()) this.toByte() else null\n}\n\ninternal fun Double.toByteExactOrNull(): Byte? {\n return if (this in Byte.MIN_VALUE.toDouble()..Byte.MAX_VALUE.toDouble()) this.toByte() else null\n}\n\ninternal fun Float.toByteExactOrNull(): Byte? {\n return if (this in Byte.MIN_VALUE.toFloat()..Byte.MAX_VALUE.toFloat()) this.toByte() else null\n}\n\ninternal fun Long.toIntExactOrNull(): Int? {\n return if (this in Int.MIN_VALUE.toLong()..Int.MAX_VALUE.toLong()) this.toInt() else null\n}\n\ninternal fun Double.toIntExactOrNull(): Int? {\n return if (this in Int.MIN_VALUE.toDouble()..Int.MAX_VALUE.toDouble()) this.toInt() else null\n}\n\ninternal fun Float.toIntExactOrNull(): Int? {\n return if (this in Int.MIN_VALUE.toFloat()..Int.MAX_VALUE.toFloat()) this.toInt() else null\n}\n\ninternal fun

```

this.toInt() else null\n}\n\ninternal fun Double.toLongExactOrNull(): Long? {\n    return if (this in
Long.MIN_VALUE.toDouble()..Long.MAX_VALUE.toDouble()) this.toLong() else null\n}\n\ninternal fun
Float.toLongExactOrNull(): Long? {\n    return if (this in
Long.MIN_VALUE.toFloat()..Long.MAX_VALUE.toFloat()) this.toLong() else null\n}\n\ninternal fun
Int.toShortExactOrNull(): Short? {\n    return if (this in Short.MIN_VALUE.toInt()..Short.MAX_VALUE.toInt())
this.toShort() else null\n}\n\ninternal fun Long.toShortExactOrNull(): Short? {\n    return if (this in
Short.MIN_VALUE.toLong()..Short.MAX_VALUE.toLong()) this.toShort() else null\n}\n\ninternal fun
Double.toShortExactOrNull(): Short? {\n    return if (this in
Short.MIN_VALUE.toDouble()..Short.MAX_VALUE.toDouble()) this.toShort() else null\n}\n\ninternal fun
Float.toShortExactOrNull(): Short? {\n    return if (this in
Short.MIN_VALUE.toFloat()..Short.MAX_VALUE.toFloat()) this.toShort() else null\n}\n\n/**\n * Returns a range
from this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this`
value, then the returned range is empty.\n */\npublic infix fun Int.until(to: Byte): IntRange {\n    return this ..
(to.toInt() - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n *
\n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun
Long.until(to: Byte): LongRange {\n    return this .. (to.toLong() - 1).toLong()\n}\n\n/**\n * Returns a range from
this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this`
value, then the returned range is empty.\n */\npublic infix fun Byte.until(to: Byte): IntRange {\n    return this.toInt() ..
(to.toInt() - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n *
\n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun
Short.until(to: Byte): IntRange {\n    return this.toInt() .. (to.toInt() - 1).toInt()\n}\n\n/**\n * Returns a range from
this value up to but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this`
value, then the returned range is empty.\n */\npublic infix fun Char.until(to: Char): CharRange {\n    if (to <= '\u0000')
return CharRange.EMPTY\n    return this .. (to - 1).toChar()\n}\n\n/**\n * Returns a range from this value up to but
excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned
range is empty.\n */\npublic infix fun Int.until(to: Int): IntRange {\n    if (to <= Int.MIN_VALUE) return
IntRange.EMPTY\n    return this .. (to - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding
the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned range is
empty.\n */\npublic infix fun Long.until(to: Int): LongRange {\n    return this .. (to.toLong() -
1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the
[to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Byte.until(to:
Int): IntRange {\n    if (to <= Int.MIN_VALUE) return IntRange.EMPTY\n    return this.toInt() .. (to -
1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the
[to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Short.until(to:
Int): IntRange {\n    if (to <= Int.MIN_VALUE) return IntRange.EMPTY\n    return this.toInt() .. (to -
1).toInt()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the
[to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Int.until(to:
Long): LongRange {\n    if (to <= Long.MIN_VALUE) return LongRange.EMPTY\n    return this.toLong() .. (to -
1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the
[to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Long.until(to:
Long): LongRange {\n    if (to <= Long.MIN_VALUE) return LongRange.EMPTY\n    return this .. (to -
1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the
[to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Byte.until(to:
Long): LongRange {\n    if (to <= Long.MIN_VALUE) return LongRange.EMPTY\n    return this.toLong() .. (to -
1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the
[to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Short.until(to:
Long): LongRange {\n    if (to <= Long.MIN_VALUE) return LongRange.EMPTY\n    return this.toLong() .. (to -
1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the

```



```

[to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Int.until(to:
Short): IntRange {\n    return this .. (to.toInt() - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to but
excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned
range is empty.\n */\npublic infix fun Long.until(to: Short): LongRange {\n    return this .. (to.toLong() -
1).toLong()\n}\n\n/**\n * Returns a range from this value up to but excluding the specified [to] value.\n * \n * If the
[to] value is less than or equal to `this` value, then the returned range is empty.\n */\npublic infix fun Byte.until(to:
Short): IntRange {\n    return this.toInt() .. (to.toInt() - 1).toInt()\n}\n\n/**\n * Returns a range from this value up to
but excluding the specified [to] value.\n * \n * If the [to] value is less than or equal to `this` value, then the returned
range is empty.\n */\npublic infix fun Short.until(to: Short): IntRange {\n    return this.toInt() .. (to.toInt() -
1).toInt()\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this
value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtLeastComparable\n */\npublic fun <T : Comparable<T>>
T.coerceAtLeast(minimumValue: T): T {\n    return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n * @return this value if it's greater
than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Byte.coerceAtLeast(minimumValue: Byte):
Byte {\n    return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less
than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue]
or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Short.coerceAtLeast(minimumValue: Short): Short {\n    return if (this < minimumValue)
minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n *
@return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n *
@sample samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Int.coerceAtLeast(minimumValue:
Int): Int {\n    return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not
less than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the
[minimumValue] or the [minimumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Long.coerceAtLeast(minimumValue: Long):
Long {\n    return if (this < minimumValue) minimumValue else this\n}\n\n/**\n * Ensures that this value is not less
than the specified [minimumValue].\n * \n * @return this value if it's greater than or equal to the [minimumValue]
or the [minimumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun Float.coerceAtLeast(minimumValue: Float): Float {\n    return if (this < minimumValue)
minimumValue else this\n}\n\n/**\n * Ensures that this value is not less than the specified [minimumValue].\n * \n *
@return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * \n *
@sample samples.comparisons.ComparableOps.coerceAtLeast\n */\npublic fun
Double.coerceAtLeast(minimumValue: Double): Double {\n    return if (this < minimumValue) minimumValue else
this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this
value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtMostComparable\n */\npublic fun <T : Comparable<T>>
T.coerceAtMost(maximumValue: T): T {\n    return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less
than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun Byte.coerceAtMost(maximumValue: Byte):
Byte {\n    return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not
greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the
[maximumValue] or the [maximumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun Short.coerceAtMost(maximumValue: Short):
Short {\n    return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not
greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the

```

```

[maximumValue] or the [maximumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun Int.coerceAtMost(maximumValue: Int): Int
{\n    return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater
than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the [maximumValue] or
the [maximumValue] otherwise.\n * \n * @sample samples.comparisons.ComparableOps.coerceAtMost\n
*/\npublic fun Long.coerceAtMost(maximumValue: Long): Long {\n    return if (this > maximumValue)
maximumValue else this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n
*\n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n
*\n * @sample samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun
Float.coerceAtMost(maximumValue: Float): Float {\n    return if (this > maximumValue) maximumValue else
this\n}\n\n/**\n * Ensures that this value is not greater than the specified [maximumValue].\n * \n * @return this
value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtMost\n */\npublic fun Double.coerceAtMost(maximumValue:
Double): Double {\n    return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this
value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range,
or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than
[maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceInComparable\n */\npublic fun
<T : Comparable<T>> T.coerceIn(minimumValue: T?, maximumValue: T?): T {\n    if (minimumValue != null
&& maximumValue != null) {\n        if (minimumValue > maximumValue) throw
IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")\n        if (this < minimumValue) return minimumValue\n        if (this >
maximumValue) return maximumValue\n    }\n    else {\n        if (minimumValue != null && this <
minimumValue) return minimumValue\n        if (maximumValue != null && this > maximumValue) return
maximumValue\n    }\n    return this\n}\n\n/**\n * Ensures that this value lies in the specified range
[minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value
is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample
samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Byte.coerceIn(minimumValue: Byte,
maximumValue: Byte): Byte {\n    if (minimumValue > maximumValue) throw
IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")\n    if (this < minimumValue) return minimumValue\n    if (this > maximumValue)
return maximumValue\n    return this\n}\n\n/**\n * Ensures that this value lies in the specified range
[minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value
is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample
samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Short.coerceIn(minimumValue: Short,
maximumValue: Short): Short {\n    if (minimumValue > maximumValue) throw
IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")\n    if (this < minimumValue) return minimumValue\n    if (this > maximumValue)
return maximumValue\n    return this\n}\n\n/**\n * Ensures that this value lies in the specified range
[minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value
is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample
samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Int.coerceIn(minimumValue: Int, maximumValue:
Int): Int {\n    if (minimumValue > maximumValue) throw IllegalArgumentOutOfRangeException("Cannot coerce value to an
empty range: maximum $maximumValue is less than minimum $minimumValue.")\n    if (this < minimumValue)
return minimumValue\n    if (this > maximumValue) return maximumValue\n    return this\n}\n\n/**\n * Ensures
that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in
the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater
than [maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceIn\n */\npublic fun
Long.coerceIn(minimumValue: Long, maximumValue: Long): Long {\n    if (minimumValue > maximumValue)

```

```

throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")\n if (this < minimumValue) return minimumValue\n if (this > maximumValue)
return maximumValue\n return this\n}\n\n/**\n * Ensures that this value lies in the specified range
[minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value
is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample
samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Float.coerceIn(minimumValue: Float,
maximumValue: Float): Float {\n if (minimumValue > maximumValue) throw
IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")\n if (this < minimumValue) return minimumValue\n if (this > maximumValue)
return maximumValue\n return this\n}\n\n/**\n * Ensures that this value lies in the specified range
[minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value
is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample
samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Double.coerceIn(minimumValue: Double,
maximumValue: Double): Double {\n if (minimumValue > maximumValue) throw
IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")\n if (this < minimumValue) return minimumValue\n if (this > maximumValue)
return maximumValue\n return this\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * \n *
@return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive`
if this value is greater than `range.endInclusive`.\n * \n * @sample
samples.comparisons.ComparableOps.coerceInFloatingPointRange\n */\n@SinceKotlin("1.1")\npublic fun <T :
Comparable<T>> T.coerceIn(range: ClosedFloatingPointRange<T>): T {\n if (range.isEmpty()) throw
IllegalArgumentException("Cannot coerce value to an empty range: $range.")\n return when {\n // this <
start equiv to this <= start && !(this >= start)\n range.lessThanOrEqualTo(this, range.start) &&
!range.lessThanOrEqualTo(range.start, this) -> range.start\n // this > end equiv to this >= end && !(this <= end)\n
range.lessThanOrEqualTo(range.endInclusive, this) && !range.lessThanOrEqualTo(this, range.endInclusive) ->
range.endInclusive\n else -> this\n }\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * \n *
@return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive`
if this value is greater than `range.endInclusive`.\n * \n * @sample
samples.comparisons.ComparableOps.coerceInComparable\n */\npublic fun <T : Comparable<T>>
T.coerceIn(range: ClosedRange<T>): T {\n if (range is ClosedFloatingPointRange) {\n return
this.coerceIn<T>(range)\n }\n if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an
empty range: $range.")\n return when {\n this < range.start -> range.start\n this > range.endInclusive ->
range.endInclusive\n else -> this\n }\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * \n *
@return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive`
if this value is greater than `range.endInclusive`.\n * \n * @sample
samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Int.coerceIn(range: ClosedRange<Int>): Int {\n if (range is ClosedFloatingPointRange) {\n
return this.coerceIn<Int>(range)\n }\n if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce
value to an empty range: $range.")\n return when {\n this < range.start -> range.start\n this >
range.endInclusive -> range.endInclusive\n else -> this\n }\n}\n\n/**\n * Ensures that this value lies in the
specified [range].\n * \n * @return this value if it's in the [range], or `range.start` if this value is less than
`range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.\n * \n * @sample
samples.comparisons.ComparableOps.coerceIn\n */\npublic fun Long.coerceIn(range: ClosedRange<Long>): Long
{\n if (range is ClosedFloatingPointRange) {\n return this.coerceIn<Long>(range)\n }\n if
(range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")\n return
when {\n this < range.start -> range.start\n this > range.endInclusive -> range.endInclusive\n else ->
this\n }\n}\n\n"/**\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt
file.\r\n */\r\n\r\n// Auto-generated file. DO NOT EDIT!\r\n\r\n\r\npackage kotlin\r\n\r\n\r\nimport

```

```

kotlin.experimental.*\r\n\r\n@Suppress("\NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")\
\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic inline class UByte @PublishedApi internal
constructor(@PublishedApi internal val data: Byte) : Comparable<UByte> {\r\n\r\n    companion object {\r\n
/**\r\n        * A constant holding the minimum value an instance of UByte can have.\r\n        */\r\n        public const
val MIN_VALUE: UByte = UByte(0)\r\n\r\n        /**\r\n            * A constant holding the maximum value an instance
of UByte can have.\r\n            */\r\n            public const val MAX_VALUE: UByte = UByte(-1)\r\n\r\n            /**\r\n
* The number of bytes used to represent an instance of UByte in a binary form.\r\n            */\r\n            public const val
SIZE_BYTES: Int = 1\r\n\r\n            /**\r\n                * The number of bits used to represent an instance of UByte in a
binary form.\r\n                */\r\n                public const val SIZE_BITS: Int = 8\r\n            }\r\n\r\n            /**\r\n
* Compares this value with the specified value for order.\r\n            * Returns zero if this value is equal to the specified other value, a
negative number if it's less than other,\r\n            * or a positive number if it's greater than other.\r\n            */\r\n
@kotlin.internal.InlineOnly\r\n            @Suppress("\OVERRIDE_BY_INLINE")\r\n            public override inline operator
fun compareTo(other: UByte): Int = this.toInt().compareTo(other.toInt())\r\n\r\n            /**\r\n                * Compares this value
with the specified value for order.\r\n                * Returns zero if this value is equal to the specified other value, a negative
number if it's less than other,\r\n                * or a positive number if it's greater than other.\r\n                */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun compareTo(other: UShort): Int =
this.toInt().compareTo(other.toInt())\r\n\r\n            /**\r\n                * Compares this value with the specified value for order.\r\n
                * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\r\n
                * or a positive number if it's greater than other.\r\n                */\r\n                @kotlin.internal.InlineOnly\r\n
            public inline operator fun compareTo(other: UInt): Int = this.toUInt().compareTo(other)\r\n\r\n            /**\r\n
                * Compares this value with the specified value for order.\r\n                * Returns zero if this value is equal to the specified other value, a negative number if
it's less than other,\r\n                * or a positive number if it's greater than other.\r\n                */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun compareTo(other: ULong): Int =
this.toULong().compareTo(other)\r\n\r\n            /**\r\n                * Adds the other value to this value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun plus(other: UByte): UInt =
this.toUInt().plus(other.toUInt())\r\n\r\n            /**\r\n                * Adds the other value to this value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun plus(other: UShort): UInt = this.toUInt().plus(other.toUInt())\r\n\r\n
            /**\r\n                * Adds the other value to this value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun plus(other: UInt): UInt =
this.toUInt().plus(other)\r\n\r\n            /**\r\n                * Adds the other value to this value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)\r\n\r\n
            /**\r\n                * Subtracts the other value from this value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun minus(other: UByte): UInt =
this.toUInt().minus(other.toUInt())\r\n\r\n            /**\r\n                * Subtracts the other value from this value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun minus(other: UShort): UInt =
this.toUInt().minus(other.toUInt())\r\n\r\n            /**\r\n                * Subtracts the other value from this value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other)\r\n\r\n
            /**\r\n                * Subtracts the other value from this value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)\r\n\r\n
            /**\r\n                * Multiplies this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun times(other: UByte): UInt =
this.toUInt().times(other.toUInt())\r\n\r\n            /**\r\n                * Multiplies this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun times(other: UShort): UInt =
this.toUInt().times(other.toUInt())\r\n\r\n            /**\r\n                * Multiplies this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun times(other: UInt): UInt = this.toUInt().times(other)\r\n\r\n
            /**\r\n                * Multiplies this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun times(other: ULong): ULong = this.toULong().times(other)\r\n\r\n
            /**\r\n                * Divides this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun div(other: UByte): UInt =
this.toUInt().div(other.toUInt())\r\n\r\n            /**\r\n                * Divides this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n            public inline operator fun div(other: UShort): UInt =
this.toUInt().div(other.toUInt())\r\n\r\n            /**\r\n                * Divides this value by the other value. */\r\n

```

```

@kotlin.internal.InlineOnly\r\n    public inline operator fun div(other: UInt): UInt = this.toUInt().div(other)\r\n    /**
Divides this value by the other value. */\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun div(other:
ULong): ULong = this.toULong().div(other)\r\n\r\n    /** Calculates the remainder of dividing this value by the
other value. */\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun rem(other: UByte): UInt =
this.toUInt().rem(other.toUInt())\r\n    /** Calculates the remainder of dividing this value by the other value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun rem(other: UShort): UInt =
this.toUInt().rem(other.toUInt())\r\n    /** Calculates the remainder of dividing this value by the other value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other)\r\n
    /** Calculates the remainder of dividing this value by the other value. */\r\n    @kotlin.internal.InlineOnly\r\n
    public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)\r\n\r\n    /** Increments this
value. */\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun inc(): UByte = UByte(data.inc())\r\n    /**
Decrements this value. */\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun dec(): UByte =
UByte(data.dec())\r\n\r\n    /** Creates a range from this value to the specified [other] value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun rangeTo(other: UByte): UIntRange =
UIntRange(this.toUInt(), other.toUInt())\r\n\r\n    /** Performs a bitwise AND operation between the two values.
*/\r\n    @kotlin.internal.InlineOnly\r\n    public inline infix fun and(other: UByte): UByte = UByte(this.data and
other.data)\r\n    /** Performs a bitwise OR operation between the two values. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline infix fun or(other: UByte): UByte = UByte(this.data or other.data)\r\n
    /** Performs a bitwise XOR operation between the two values. */\r\n    @kotlin.internal.InlineOnly\r\n    public
inline infix fun xor(other: UByte): UByte = UByte(this.data xor other.data)\r\n    /** Inverts the bits in this value.
*/\r\n    @kotlin.internal.InlineOnly\r\n    public inline fun inv(): UByte = UByte(data.inv())\r\n\r\n    /**
Converts this [UByte] value to [Byte].\r\n    * If this value is less than or equals to [Byte.MAX_VALUE],
the resulting `Byte` value represents\r\n    * the same numerical value as this `UByte`. Otherwise the result is
negative.\r\n    * The resulting `Byte` value has the same binary representation as this `UByte` value.\r\n
    */\r\n    @kotlin.internal.InlineOnly\r\n    public inline fun toByte(): Byte = data\r\n    /**
Converts this [UByte] value to [Short].\r\n    * The resulting `Short` value represents the same numerical value as this
`UByte`.\r\n    * The least significant 8 bits of the resulting `Short` value are the same as the bits of this
`UByte` value,\r\n    * whereas the most significant 8 bits are filled with zeros.\r\n
    */\r\n    @kotlin.internal.InlineOnly\r\n    public inline fun toShort(): Short = data.toShort() and 0xFF\r\n    /**
Converts this [UByte] value to [Int].\r\n    * The resulting `Int` value represents the same numerical value as
this `UByte`.\r\n    * The least significant 8 bits of the resulting `Int` value are the same as the bits of this
`UByte` value,\r\n    * whereas the most significant 24 bits are filled with zeros.\r\n
    */\r\n    @kotlin.internal.InlineOnly\r\n    public inline fun toInt(): Int = data.toInt() and 0xFF\r\n    /**
Converts this [UByte] value to [Long].\r\n    * The resulting `Long` value represents the same numerical value as this
`UByte`.\r\n    * The least significant 8 bits of the resulting `Long` value are the same as the bits of this
`UByte` value,\r\n    * whereas the most significant 56 bits are filled with zeros.\r\n
    */\r\n    @kotlin.internal.InlineOnly\r\n    public inline fun toLong(): Long = data.toLong() and 0xFF\r\n\r\n    /** Returns
this value. */\r\n    @kotlin.internal.InlineOnly\r\n    public inline fun toUByte(): UByte = this\r\n    /**
Converts this [UByte] value to [UShort].\r\n    * The resulting `UShort` value represents the same numerical
value as this `UByte`.\r\n    * The least significant 8 bits of the resulting `UShort` value are the same as the
bits of this `UByte` value,\r\n    * whereas the most significant 8 bits are filled with zeros.\r\n
    */\r\n    @kotlin.internal.InlineOnly\r\n    public inline fun toUShort(): UShort = UShort(data.toShort() and 0xFF)\r\n
    /**
Converts this [UByte] value to [UInt].\r\n    * The resulting `UInt` value represents the same
numerical value as this `UByte`.\r\n    * The least significant 8 bits of the resulting `UInt` value are the same
as the bits of this `UByte` value,\r\n    * whereas the most significant 24 bits are filled with zeros.\r\n
    */\r\n    @kotlin.internal.InlineOnly\r\n    public inline fun toUInt(): UInt = UInt(data.toInt() and 0xFF)\r\n    /**
Converts this [UByte] value to [ULong].\r\n    * The resulting `ULong` value represents the same numerical
value as this `UByte`.\r\n    * The least significant 8 bits of the resulting `ULong` value are the same as the

```

```

bits of this `UByte` value,
 * whereas the most significant 56 bits are filled with zeros.
 */
@kotlin.internal.InlineOnly
public inline fun toULong(): ULong = ULong(data.toLong() and 0xFF)
/**
 * Converts this [UByte] value to [Float].
 * The resulting `Float` value represents the same
numerical value as this `UByte`.
 */
@kotlin.internal.InlineOnly
public inline fun toFloat(): Float =
this.toInt().toFloat()
/**
 * Converts this [UByte] value to [Double].
 * The resulting
`Double` value represents the same numerical value as this `UByte`.
 */
@kotlin.internal.InlineOnly
public inline fun toDouble(): Double = this.toInt().toDouble()
public override fun toString(): String =
toInt().toString()
}
/**
 * Converts this [Byte] value to [UByte].
 * If this value is positive,
the resulting `UByte` value represents the same numerical value as this `Byte`.
 * The resulting `UByte`
value has the same binary representation as this `Byte` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
Byte.toUByte(): UByte = UByte(this)
/**
 * Converts this [Short] value to [UByte].
 * If this value is
positive and less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents
 * the same
numerical value as this `Short`.
 * The resulting `UByte` value is represented by the least significant 8 bits
of this `Short` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
Short.toUByte(): UByte = UByte(this.toByte())
/**
 * Converts this [Int] value to [UByte].
 * If this
value is positive and less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents
 * the
same numerical value as this `Int`.
 * The resulting `UByte` value is represented by the least significant 8
bits of this `Int` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
Int.toUByte(): UByte = UByte(this.toByte())
/**
 * Converts this [Long] value to [UByte].
 * If this
value is positive and less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents
 * the
same numerical value as this `Long`.
 * The resulting `UByte` value is represented by the least significant 8
bits of this `Long` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
Long.toUByte(): UByte = UByte(this.toByte())
"/**
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin
Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.
 */
// Auto-generated file. DO NOT EDIT!
package
kotlin
kotlin.nimport
kotlin.experimental
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class UInt @PublishedApi internal
constructor(@PublishedApi internal val data: Int) : Comparable<UInt> {
    companion object {
        /**
         * A constant holding the minimum value an instance of UInt can have.
         */
        public const
val MIN_VALUE: UInt = UInt(0)
        /**
         * A constant holding the maximum value an instance of
UInt can have.
         */
        public const val MAX_VALUE: UInt = UInt(-1)
        /**
         * The
number of bytes used to represent an instance of UInt in a binary form.
         */
        public const val
SIZE_BYTES: Int = 4
        /**
         * The number of bits used to represent an instance of UInt in a binary
form.
         */
        public const val SIZE_BITS: Int = 32
    }
    /**
     * Compares this value with
the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative
number if it's less than other,
     * or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: UByte): Int =
this.compareTo(other.toUInt())
/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
 * or a
positive number if it's greater than other.
 */
@kotlin.internal.InlineOnly
public inline operator fun
compareTo(other: UShort): Int = this.compareTo(other.toUInt())
/**
 * Compares this value with the
specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if
it's less than other,
 * or a positive number if it's greater than other.
 */

```

```

@kotlin.internal.InlineOnly\r\n  @Suppress("OVERRIDE_BY_INLINE")\r\n  public override inline operator
fun compareTo(other: UInt): Int = uintCompare(this.data, other.data)\r\n\r\n  /**\r\n   * Compares this value with
the specified value for order.\r\n   * Returns zero if this value is equal to the specified other value, a negative
number if it's less than other,\r\n   * or a positive number if it's greater than other.\r\n   */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun compareTo(other: ULong): Int =
this.toULong().compareTo(other)\r\n\r\n  /** Adds the other value to this value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun plus(other: UByte): UInt = this.plus(other.toUInt())\r\n
/** Adds the other value to this value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun plus(other: UShort): UInt = this.plus(other.toUInt())\r\n
/** Adds the other value to this value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun plus(other: UInt): UInt =
UInt(this.data.plus(other.data))\r\n
/** Adds the other value to this value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)\r\n\r\n
/** Subtracts the other
value from this value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun minus(other: UByte): UInt
= this.minus(other.toUInt())\r\n
/** Subtracts the other value from this value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun minus(other: UShort): UInt =
this.minus(other.toUInt())\r\n
/** Subtracts the other value from this value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun minus(other: UInt): UInt = UInt(this.data.minus(other.data))\r\n
/** Subtracts the other
value from this value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun minus(other: ULong):
ULong = this.toULong().minus(other)\r\n\r\n
/** Multiplies this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun times(other: UByte): UInt =
this.times(other.toUInt())\r\n
/** Multiplies this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun times(other: UShort): UInt = this.times(other.toUInt())\r\n
/** Multiplies this value by
the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun times(other: UInt): UInt =
UInt(this.data.times(other.data))\r\n
/** Multiplies this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun times(other: ULong): ULong =
this.toULong().times(other)\r\n\r\n
/** Divides this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun div(other: UByte): UInt = this.div(other.toUInt())\r\n
/** Divides this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun div(other: UShort): UInt = this.div(other.toUInt())\r\n
/** Divides this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun div(other: UInt): UInt = uintDivide(this, other)\r\n
/**
Divides this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun div(other:
ULong): ULong = this.toULong().div(other)\r\n\r\n
/** Calculates the remainder of dividing this value by the
other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun rem(other: UByte): UInt =
this.rem(other.toUInt())\r\n
/** Calculates the remainder of dividing this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun rem(other: UShort): UInt = this.rem(other.toUInt())\r\n
/** Calculates the remainder of dividing this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun rem(other: UInt): UInt = uintRemainder(this, other)\r\n
/** Calculates the remainder of
dividing this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun rem(other:
ULong): ULong = this.toULong().rem(other)\r\n\r\n
/** Increments this value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun inc(): UInt = UInt(data.inc())\r\n
/** Decrements this
value. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline operator fun dec(): UInt = UInt(data.dec())\r\n\r\n
/** Creates a range from this value to the specified [other] value. */\r\n
@kotlin.internal.InlineOnly\r\n  public
inline operator fun rangeTo(other: UInt): UIntRange = UIntRange(this, other)\r\n\r\n
/** Shifts this value left by
the [bitCount] number of bits. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline infix fun shl(bitCount: Int):
UInt = UInt(data shl bitCount)\r\n
/** Shifts this value right by the [bitCount] number of bits, filling the leftmost
bits with zeros. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline infix fun shr(bitCount: Int): UInt = UInt(data
ushr bitCount)\r\n
/** Performs a bitwise AND operation between the two values. */\r\n
@kotlin.internal.InlineOnly\r\n  public inline infix fun and(other: UInt): UInt = UInt(this.data and other.data)\r\n

```

```

/** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly public inline
infix fun or(other: UInt): UInt = UInt(this.data or other.data)
/** Performs a bitwise XOR operation between
the two values. */
@kotlin.internal.InlineOnly public inline infix fun xor(other: UInt): UInt =
UInt(this.data xor other.data)
/** Inverts the bits in this value. */
@kotlin.internal.InlineOnly public inline
fun inv(): UInt = UInt(data.inv())
/** Converts this [UInt] value to [Byte].
If this value is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents
the same numerical value as this `UInt`.
The resulting `Byte` value is represented by the least significant 8 bits
of this `UInt` value.
Note that the resulting `Byte` value may be negative. */
@kotlin.internal.InlineOnly public inline fun toByte(): Byte = data.toByte()
/** Converts this
[UInt] value to [Short].
If this value is less than or equals to [Short.MAX_VALUE], the resulting
`Short` value represents
the same numerical value as this `UInt`.
The resulting `Short` value
is represented by the least significant 16 bits of this `UInt` value.
Note that the resulting `Short` value may
be negative. */
@kotlin.internal.InlineOnly public inline fun toShort(): Short = data.toShort()
/** Converts this [UInt] value to [Int].
If this value is less than or equals to
[Int.MAX_VALUE], the resulting `Int` value represents
the same numerical value as this `UInt`. Otherwise
the result is negative.
The resulting `Int` value has the same binary representation as this `UInt`
value. */
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data
/** Converts
this [UInt] value to [Long].
The resulting `Long` value represents the same numerical value as this
`UInt`.
The least significant 32 bits of the resulting `Long` value are the same as the bits of this
`UInt` value,
whereas the most significant 32 bits are filled with zeros. */
@kotlin.internal.InlineOnly public inline fun toLong(): Long = data.toLong() and 0xFFFF_FFFF
/** Converts this [UInt] value to [UByte].
If this value is less than or equals to
[UByte.MAX_VALUE], the resulting `UByte` value represents
the same numerical value as this `UInt`.
The resulting `UByte` value is represented by the least significant 8 bits of this `UInt` value. */
@kotlin.internal.InlineOnly public inline fun toUByte(): UByte = data.toUByte()
/** Converts
this [UInt] value to [UShort].
If this value is less than or equals to [UShort.MAX_VALUE], the
resulting `UShort` value represents
the same numerical value as this `UInt`.
The resulting
`UShort` value is represented by the least significant 16 bits of this `UInt` value. */
@kotlin.internal.InlineOnly public inline fun toUShort(): UShort = data.toUShort()
/** Returns this
value. */
@kotlin.internal.InlineOnly public inline fun toUInt(): UInt = this
/** Converts this
[UInt] value to [ULong].
The resulting `ULong` value represents the same numerical value as this
`UInt`.
The least significant 32 bits of the resulting `ULong` value are the same as the bits of this
`UInt` value,
whereas the most significant 32 bits are filled with zeros. */
@kotlin.internal.InlineOnly public inline fun toULong(): ULong = ULong(data.toLong() and
0xFFFF_FFFF)
/** Converts this [UInt] value to [Float].
The resulting value is the
closest `Float` to this `UInt` value.
In case when this `UInt` value is exactly between two `Float`s,
the one with zero at least significant bit of mantissa is selected. */
@kotlin.internal.InlineOnly public inline fun toFloat(): Float = this.toDouble().toFloat()
/** Converts this [UInt] value to
[Double].
The resulting `Double` value represents the same numerical value as this `UInt`. */
@kotlin.internal.InlineOnly public inline fun toDouble(): Double = UIntToDouble(data)
public override fun toString(): String = toLong().toString()
/** Converts this [Byte] value to [UInt].
If this value is positive, the resulting `UInt` value represents the same numerical value as this `Byte`.
The least significant 8 bits of the resulting `UInt` value are the same as the bits of this `Byte` value,
whereas
the most significant 24 bits are filled with the sign bit of this value. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly public inline fun
Byte.toUInt(): UInt = UInt(this.toInt())
/** Converts this [Short] value to [UInt].
If this value is
positive, the resulting `UInt` value represents the same numerical value as this `Short`.
The least
significant 16 bits of the resulting `UInt` value are the same as the bits of this `Short` value,
whereas the most

```



```

significant 16 bits are filled with the sign bit of this value.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
Short.toUInt(): UInt = UInt(this.toInt())\r\n/**\r\n * Converts this [Int] value to [UInt].\r\n * If this value is
positive, the resulting `UInt` value represents the same numerical value as this `Int`.\r\n * The resulting `UInt`
value has the same binary representation as this `Int` value.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
Int.toUInt(): UInt = UInt(this)\r\n/**\r\n * Converts this [Long] value to [UInt].\r\n * If this value is positive
and less than or equals to [UInt.MAX_VALUE], the resulting `UInt` value represents\r\n * the same numerical
value as this `Long`.\r\n * The resulting `UInt` value is represented by the least significant 32 bits of this
`Long` value.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
Long.toUInt(): UInt = UInt(this.toInt())\r\n\r\n/**\r\n * Converts this [Float] value to [UInt].\r\n * The
fractional part, if any, is rounded down.\r\n * Returns zero if this `Float` value is negative or `NaN`,
[UInt.MAX_VALUE] if it's bigger than `UInt.MAX_VALUE`.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
Float.toUInt(): UInt = doubleToUInt(this.toDouble())\r\n/**\r\n * Converts this [Double] value to [UInt].\r\n *
The fractional part, if any, is rounded down.\r\n * Returns zero if this `Double` value is negative or `NaN`,
[UInt.MAX_VALUE] if it's bigger than `UInt.MAX_VALUE`.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
Double.toUInt(): UInt = doubleToUInt(this)\r\n", "\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n *\r\n\r\n// Auto-generated file. DO NOT EDIT!\r\n\r\npackage
kotlin\r\n\r\nimport
kotlin.experimental.*\r\n\r\n@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")\r\n
\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic inline class UShort @PublishedApi internal
constructor(@PublishedApi internal val data: Short) : Comparable<UShort> {\r\n\r\n    companion object {\r\n
/**\r\n     * A constant holding the minimum value an instance of UShort can have.\r\n     *\r\n     public
const val MIN_VALUE: UShort = UShort(0)\r\n\r\n     /**\r\n     * A constant holding the maximum value an
instance of UShort can have.\r\n     *\r\n     public const val MAX_VALUE: UShort = UShort(-1)\r\n\r\n
/**\r\n     * The number of bytes used to represent an instance of UShort in a binary form.\r\n     *\r\n
public const val SIZE_BYTES: Int = 2\r\n\r\n     /**\r\n     * The number of bits used to represent an instance of
UShort in a binary form.\r\n     *\r\n     public const val SIZE_BITS: Int = 16\r\n    }\r\n\r\n    /**\r\n     *
Compares this value with the specified value for order.\r\n     * Returns zero if this value is equal to the specified
other value, a negative number if it's less than other,\r\n     * or a positive number if it's greater than other.\r\n
*\r\n     @kotlin.internal.InlineOnly\r\n     public inline operator fun compareTo(other: UByte): Int =
this.toInt().compareTo(other.toInt())\r\n\r\n     /**\r\n     * Compares this value with the specified value for order.\r\n
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\r\n
     * or a positive number if it's greater than other.\r\n     *\r\n     @kotlin.internal.InlineOnly\r\n
public override inline operator fun compareTo(other: UShort): Int
= this.toInt().compareTo(other.toInt())\r\n\r\n     /**\r\n     * Compares this value with the specified value for
order.\r\n     * Returns zero if this value is equal to the specified other value, a negative number if it's less than
other,\r\n     * or a positive number if it's greater than other.\r\n     *\r\n     @kotlin.internal.InlineOnly\r\n
public inline operator fun compareTo(other: UInt): Int = this.toUInt().compareTo(other)\r\n\r\n     /**\r\n     *
Compares this value with the specified value for order.\r\n     * Returns zero if this value is equal to the specified
other value, a negative number if it's less than other,\r\n     * or a positive number if it's greater than other.\r\n
     *\r\n     @kotlin.internal.InlineOnly\r\n     public inline operator fun compareTo(other: ULong): Int =
this.toULong().compareTo(other)\r\n\r\n     /**\r\n     * Adds the other value to this value. *\r\n\r\n
     @kotlin.internal.InlineOnly\r\n     public inline operator fun plus(other: UByte): UInt =

```

```

this.toUInt().plus(other.toUInt())\r\n /** Adds the other value to this value. */\r\n @kotlin.internal.InlineOnly\r\n
    public inline operator fun plus(other: UShort): UInt = this.toUInt().plus(other.toUInt())\r\n /** Adds the other
value to this value. */\r\n @kotlin.internal.InlineOnly\r\n    public inline operator fun plus(other: UInt): UInt =
this.toUInt().plus(other)\r\n /** Adds the other value to this value. */\r\n @kotlin.internal.InlineOnly\r\n    public
inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)\r\n\r\n /** Subtracts the other value
from this value. */\r\n @kotlin.internal.InlineOnly\r\n    public inline operator fun minus(other: UByte): UInt =
this.toUInt().minus(other.toUInt())\r\n /** Subtracts the other value from this value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun minus(other: UShort): UInt =
this.toUInt().minus(other.toUInt())\r\n /** Subtracts the other value from this value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other)\r\n
/** Subtracts the other value from this value. */\r\n @kotlin.internal.InlineOnly\r\n    public inline operator fun
minus(other: ULong): ULong = this.toULong().minus(other)\r\n\r\n /** Multiplies this value by the other value.
*/\r\n @kotlin.internal.InlineOnly\r\n    public inline operator fun times(other: UByte): UInt =
this.toUInt().times(other.toUInt())\r\n /** Multiplies this value by the other value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun times(other: UShort): UInt =
this.toUInt().times(other.toUInt())\r\n /** Multiplies this value by the other value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun times(other: UInt): UInt = this.toUInt().times(other)\r\n
/** Multiplies this value by the other value. */\r\n @kotlin.internal.InlineOnly\r\n    public inline operator fun
times(other: ULong): ULong = this.toULong().times(other)\r\n\r\n /** Divides this value by the other value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun div(other: UByte): UInt =
this.toUInt().div(other.toUInt())\r\n /** Divides this value by the other value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun div(other: UShort): UInt =
this.toUInt().div(other.toUInt())\r\n /** Divides this value by the other value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun div(other: UInt): UInt = this.toUInt().div(other)\r\n
/** Divides this value by the other value. */\r\n @kotlin.internal.InlineOnly\r\n    public inline operator fun div(other:
ULong): ULong = this.toULong().div(other)\r\n\r\n /** Calculates the remainder of dividing this value by the
other value. */\r\n @kotlin.internal.InlineOnly\r\n    public inline operator fun rem(other: UByte): UInt =
this.toUInt().rem(other.toUInt())\r\n /** Calculates the remainder of dividing this value by the other value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun rem(other: UShort): UInt =
this.toUInt().rem(other.toUInt())\r\n /** Calculates the remainder of dividing this value by the other value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other)\r\n
/** Calculates the remainder of dividing this value by the other value. */\r\n @kotlin.internal.InlineOnly\r\n
    public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)\r\n\r\n /** Increments this
value. */\r\n @kotlin.internal.InlineOnly\r\n    public inline operator fun inc(): UShort = UShort(data.inc())\r\n
/** Decrements this value. */\r\n @kotlin.internal.InlineOnly\r\n    public inline operator fun dec(): UShort =
UShort(data.dec())\r\n\r\n /** Creates a range from this value to the specified [other] value. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())\r\n\r\n /** Performs a bitwise AND operation between the two values.
*/\r\n @kotlin.internal.InlineOnly\r\n    public inline infix fun and(other: UShort): UShort = UShort(this.data and
other.data)\r\n /** Performs a bitwise OR operation between the two values. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline infix fun or(other: UShort): UShort = UShort(this.data or
other.data)\r\n /** Performs a bitwise XOR operation between the two values. */\r\n
    @kotlin.internal.InlineOnly\r\n    public inline infix fun xor(other: UShort): UShort = UShort(this.data xor
other.data)\r\n /** Inverts the bits in this value. */\r\n @kotlin.internal.InlineOnly\r\n    public inline fun inv():
UShort = UShort(data.inv())\r\n\r\n /** * Converts this [UShort] value to [Byte].\r\n * If this value
is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents\r\n * the same numerical
value as this `UShort`.\r\n * The resulting `Byte` value is represented by the least significant 8 bits of this
`UShort` value.\r\n * Note that the resulting `Byte` value may be negative.\r\n */\r\n

```

```

@kotlin.internal.InlineOnly\r\n public inline fun toByte(): Byte = data.toByte()\r\n /**\r\n * Converts this
[UShort] value to [Short].\r\n *\r\n * If this value is less than or equals to [Short.MAX_VALUE], the resulting
`Short` value represents\r\n * the same numerical value as this `UShort`. Otherwise the result is negative.\r\n
*\r\n * The resulting `Short` value has the same binary representation as this `UShort` value.\r\n */\r\n
@kotlin.internal.InlineOnly\r\n public inline fun toShort(): Short = data\r\n /**\r\n * Converts this [UShort]
value to [Int].\r\n *\r\n * The resulting `Int` value represents the same numerical value as this `UShort`.\r\n
*\r\n * The least significant 16 bits of the resulting `Int` value are the same as the bits of this `UShort` value,\r\n
* whereas the most significant 16 bits are filled with zeros.\r\n */\r\n
@kotlin.internal.InlineOnly\r\n public
inline fun toInt(): Int = data.toInt() and 0xFFFF\r\n /**\r\n * Converts this [UShort] value to [Long].\r\n *\r\n
* The resulting `Long` value represents the same numerical value as this `UShort`.\r\n *\r\n * The least
significant 16 bits of the resulting `Long` value are the same as the bits of this `UShort` value,\r\n * whereas the
most significant 48 bits are filled with zeros.\r\n */\r\n
@kotlin.internal.InlineOnly\r\n public inline fun
toLong(): Long = data.toLong() and 0xFFFF\r\n\r\n /**\r\n * Converts this [UShort] value to [UByte].\r\n
*\r\n * If this value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents\r\n
* the same numerical value as this `UShort`.\r\n *\r\n * The resulting `UByte` value is represented by the least
significant 8 bits of this `UShort` value.\r\n */\r\n
@kotlin.internal.InlineOnly\r\n public inline fun toUByte():
UByte = data.toUByte()\r\n /** Returns this value. */\r\n
@kotlin.internal.InlineOnly\r\n public inline fun
toUShort(): UShort = this\r\n /**\r\n * Converts this [UShort] value to [UInt].\r\n *\r\n * The resulting
`UInt` value represents the same numerical value as this `UShort`.\r\n *\r\n * The least significant 16 bits of the
resulting `UInt` value are the same as the bits of this `UShort` value,\r\n * whereas the most significant 16 bits are
filled with zeros.\r\n */\r\n
@kotlin.internal.InlineOnly\r\n public inline fun toUInt(): UInt = UInt(data.toInt()
and 0xFFFF)\r\n /**\r\n * Converts this [UShort] value to [ULong].\r\n *\r\n * The resulting `ULong`
value represents the same numerical value as this `UShort`.\r\n *\r\n * The least significant 16 bits of the
resulting `ULong` value are the same as the bits of this `UShort` value,\r\n * whereas the most significant 48 bits
are filled with zeros.\r\n */\r\n
@kotlin.internal.InlineOnly\r\n public inline fun toULong(): ULong =
ULong(data.toLong() and 0xFFFF)\r\n\r\n /**\r\n * Converts this [UShort] value to [Float].\r\n *\r\n * The
resulting `Float` value represents the same numerical value as this `UShort`.\r\n */\r\n
@kotlin.internal.InlineOnly\r\n public inline fun toFloat(): Float = this.toInt().toFloat()\r\n /**\r\n * Converts
this [UShort] value to [Double].\r\n *\r\n * The resulting `Double` value represents the same numerical value as
this `UShort`.\r\n */\r\n
@kotlin.internal.InlineOnly\r\n public inline fun toDouble(): Double =
this.toInt().toDouble()\r\n\r\n public override fun toString(): String = toInt().toString()\r\n\r\n /**\r\n *
Converts this [Byte] value to [UShort].\r\n *\r\n * If this value is positive, the resulting `UShort` value represents
the same numerical value as this `Byte`.\r\n *\r\n * The least significant 8 bits of the resulting `UShort` value are the
same as the bits of this `Byte` value,\r\n * whereas the most significant 8 bits are filled with the sign bit of this
value.\r\n */\r\n
@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic
inline fun Byte.toUShort(): UShort = UShort(this.toShort())\r\n\r\n /**\r\n * Converts this [Short] value to [UShort].\r\n
*\r\n * If this value is positive, the resulting `UShort` value represents the same numerical value as this `Short`.\r\n
*\r\n * The resulting `UShort` value has the same binary representation as this `Short` value.\r\n
*/\r\n
@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
Short.toUShort(): UShort = UShort(this)\r\n\r\n /**\r\n * Converts this [Int] value to [UShort].\r\n *\r\n * If this value is
positive and less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents\r\n * the same
numerical value as this `Int`.\r\n *\r\n * The resulting `UShort` value is represented by the least significant 16 bits of
this `Int` value.\r\n
*/\r\n
@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
Int.toUShort(): UShort = UShort(this.toShort())\r\n\r\n /**\r\n * Converts this [Long] value to [UShort].\r\n *\r\n * If
this value is positive and less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents\r\n
* the same numerical value as this `Long`.\r\n *\r\n * The resulting `UShort` value is represented by the least
significant 16 bits of this `Long` value.\r\n
*/

```

```

*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
Long.toUShort(): UShort = UShort(this.toShort())\r\n", "/*\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n */\r\n\r\n// Auto-generated file. DO NOT EDIT!\r\n\r\npackage
kotlin.ranges\r\n\r\n/**\r\n * A range of values of type `Char`.\r\n */\r\npublic class CharRange(start: Char,
endInclusive: Char) : CharProgression(start, endInclusive, 1), ClosedRange<Char> {\r\n    override val start: Char
get() = first\r\n    override val endInclusive: Char get() = last\r\n\r\n    override fun contains(value: Char): Boolean =
first <= value && value <= last\r\n\r\n    override fun isEmpty(): Boolean = first > last\r\n\r\n    override fun
equals(other: Any?): Boolean =\r\n        other is CharRange && (isEmpty() && other.isEmpty()) ||\r\n        first ==
other.first && last == other.last\r\n\r\n    override fun hashCode(): Int =\r\n        if (isEmpty()) -1 else (31 *
first.toInt() + last.toInt())\r\n\r\n    override fun toString(): String = "$first..$last"\r\n\r\n    companion object {\r\n
        /** An empty range of values of type Char. */\r\n        public val EMPTY: CharRange = CharRange(1.toChar(),
0.toChar())\r\n    }\r\n}\r\n\r\n\r\n/**\r\n * A range of values of type `Int`.\r\n */\r\npublic class IntRange(start: Int,
endInclusive: Int) : IntProgression(start, endInclusive, 1), ClosedRange<Int> {\r\n    override val start: Int get() =
first\r\n    override val endInclusive: Int get() = last\r\n\r\n    override fun contains(value: Int): Boolean = first <=
value && value <= last\r\n\r\n    override fun isEmpty(): Boolean = first > last\r\n\r\n    override fun equals(other:
Any?): Boolean =\r\n        other is IntRange && (isEmpty() && other.isEmpty()) ||\r\n        first == other.first &&
last == other.last\r\n\r\n    override fun hashCode(): Int =\r\n        if (isEmpty()) -1 else (31 * first + last)\r\n\r\n
    override fun toString(): String = "$first..$last"\r\n\r\n    companion object {\r\n        /** An empty range of values
of type Int. */\r\n        public val EMPTY: IntRange = IntRange(1, 0)\r\n    }\r\n}\r\n\r\n\r\n/**\r\n * A range of values
of type `Long`.\r\n */\r\npublic class LongRange(start: Long, endInclusive: Long) : LongProgression(start,
endInclusive, 1), ClosedRange<Long> {\r\n    override val start: Long get() = first\r\n    override val endInclusive:
Long get() = last\r\n\r\n    override fun contains(value: Long): Boolean = first <= value && value <= last\r\n\r\n
    override fun isEmpty(): Boolean = first > last\r\n\r\n    override fun equals(other: Any?): Boolean =\r\n        other is
LongRange && (isEmpty() && other.isEmpty()) ||\r\n        first == other.first && last == other.last\r\n\r\n
    override fun hashCode(): Int =\r\n        if (isEmpty()) -1 else (31 * (first xor (first ushr 32)) + (last xor (last ushr
32))).toInt()\r\n\r\n    override fun toString(): String = "$first..$last"\r\n\r\n    companion object {\r\n        /** An
empty range of values of type Long. */\r\n        public val EMPTY: LongRange = LongRange(1, 0)\r\n    }\r\n}\r\n}\r\n\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n *
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n*\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("CollectionsKt")\r\n\r\npackage
kotlin.collections\r\n\r\nimport kotlin.contracts.*\r\n\r\ninternal object EmptyIterator : ListIterator<Nothing> {\r\n
    override fun hasNext(): Boolean = false\r\n    override fun hasPrevious(): Boolean = false\r\n    override fun
nextIndex(): Int = 0\r\n    override fun previousIndex(): Int = -1\r\n    override fun next(): Nothing = throw
NoSuchElementException()\r\n    override fun previous(): Nothing = throw
NoSuchElementException()\r\n}\r\n\r\n\r\ninternal object EmptyList : List<Nothing>, Serializable, RandomAccess
{\r\n    private const val serialVersionUID: Long = -7390468764508069838L\r\n\r\n    override fun equals(other:
Any?): Boolean = other is List<*> && other.isEmpty()\r\n    override fun hashCode(): Int = 1\r\n    override fun
toString(): String = "[]"\r\n\r\n    override val size: Int get() = 0\r\n    override fun isEmpty(): Boolean = true\r\n
    override fun contains(element: Nothing): Boolean = false\r\n    override fun containsAll(elements:
Collection<Nothing>): Boolean = elements.isEmpty()\r\n\r\n    override fun get(index: Int): Nothing = throw
IndexOutOfBoundsException("Empty list doesn't contain element at index $index.")\r\n    override fun
indexOf(element: Nothing): Int = -1\r\n    override fun lastIndexOf(element: Nothing): Int = -1\r\n\r\n    override fun
iterator(): Iterator<Nothing> = EmptyIterator\r\n    override fun listIterator(): ListIterator<Nothing> =
EmptyIterator\r\n    override fun listIterator(index: Int): ListIterator<Nothing> {\r\n        if (index != 0) throw
IndexOutOfBoundsException("Index: $index")\r\n        return EmptyIterator\r\n    }\r\n\r\n    override fun
subList(fromIndex: Int, toIndex: Int): List<Nothing> {\r\n        if (fromIndex == 0 && toIndex == 0) return this\r\n
        throw IndexOutOfBoundsException("fromIndex: $fromIndex, toIndex: $toIndex")\r\n    }\r\n\r\n    private fun

```

```

readResolve(): Any = EmptyList\r\n}\r\n\r\ninternal fun <T> Array<out T>.asCollection(): Collection<T> =
ArrayAsCollection(this, isVarargs = false)\r\n\r\nprivate class ArrayAsCollection<T>(val values: Array<out T>, val
isVarargs: Boolean) : Collection<T> {\r\n    override val size: Int get() = values.size\r\n    override fun isEmpty():
Boolean = values.isEmpty()\r\n    override fun contains(element: T): Boolean = values.contains(element)\r\n
override fun containsAll(elements: Collection<T>): Boolean = elements.all { contains(it) }\r\n    override fun
iterator(): Iterator<T> = values.iterator()\r\n    // override hidden to Array implementation to prevent copying of
values array\r\n    public fun toArray(): Array<out Any?> =
values.copyToArrayOfAny(isVarargs)\r\n}\r\n\r\n/**\r\n * Returns an empty read-only list. The returned list is
serializable (JVM).\r\n * @sample samples.collections.Collections.Lists.emptyReadOnlyList\r\n */\r\npublic fun
<T> emptyList(): List<T> = EmptyList\r\n\r\n/**\r\n * Returns a new read-only list of given elements. The returned
list is serializable (JVM).\r\n * @sample samples.collections.Collections.Lists.readOnlyList\r\n */\r\npublic fun <T>
listOf(vararg elements: T): List<T> = if (elements.size > 0) elements.asList() else emptyList()\r\n\r\n/**\r\n *
Returns an empty read-only list. The returned list is serializable (JVM).\r\n * @sample
samples.collections.Collections.Lists.emptyReadOnlyList\r\n */\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
<T> listOf(): List<T> = emptyList()\r\n\r\n/**\r\n * Returns an empty new [MutableList].\r\n * @sample
samples.collections.Collections.Lists.emptyMutableList\r\n
*/\r\n@SinceKotlin("1.1")\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T> mutableListOf():
MutableList<T> = ArrayList()\r\n\r\n/**\r\n * Returns an empty new [ArrayList].\r\n * @sample
samples.collections.Collections.Lists.emptyArrayList\r\n
*/\r\n@SinceKotlin("1.1")\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T> arrayListOf(): ArrayList<T> =
ArrayList()\r\n\r\n/**\r\n * Returns a new [MutableList] with the given elements.\r\n * @sample
samples.collections.Collections.Lists.mutableList\r\n */\r\npublic fun <T> mutableListOf(vararg elements: T):
MutableList<T> =\r\n    if (elements.size == 0) ArrayList() else ArrayList(ArrayAsCollection(elements, isVarargs =
true))\r\n\r\n/**\r\n * Returns a new [ArrayList] with the given elements.\r\n * @sample
samples.collections.Collections.Lists.arrayList\r\n */\r\npublic fun <T> arrayListOf(vararg elements: T):
ArrayList<T> =\r\n    if (elements.size == 0) ArrayList() else ArrayList(ArrayAsCollection(elements, isVarargs =
true))\r\n\r\n/**\r\n * Returns a new read-only list either of single given element, if it is not null, or empty list if the
element is null. The returned list is serializable (JVM).\r\n * @sample
samples.collections.Collections.Lists.listOfNotNull\r\n */\r\npublic fun <T : Any> listOfNotNull(element: T?):
List<T> = if (element != null) listOf(element) else emptyList()\r\n\r\n/**\r\n * Returns a new read-only list only of
those given elements, that are not null. The returned list is serializable (JVM).\r\n * @sample
samples.collections.Collections.Lists.listOfNotNull\r\n */\r\npublic fun <T : Any> listOfNotNull(vararg elements:
T?): List<T> = elements.filterNotNull()\r\n\r\n/**\r\n * Creates a new read-only list with the specified [size], where
each element is calculated by calling the specified\r\n * [init] function.\r\n * The function [init] is called for
each list element sequentially starting from the first one.\r\n * It should return the value for a list element given its
index.\r\n * @sample samples.collections.Collections.Lists.readOnlyListFromInitializer\r\n
*/\r\n@SinceKotlin("1.1")\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T> List(size: Int, init: (index: Int) -
> T): List<T> = MutableList(size, init)\r\n\r\n/**\r\n * Creates a new mutable list with the specified [size], where
each element is calculated by calling the specified\r\n * [init] function.\r\n * The function [init] is called for
each list element sequentially starting from the first one.\r\n * It should return the value for a list element given its
index.\r\n * @sample samples.collections.Collections.Lists.mutableListFromInitializer\r\n
*/\r\n@SinceKotlin("1.1")\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T> MutableList(size: Int, init:
(index: Int) -> T): MutableList<T> {\r\n    val list = ArrayList<T>(size)\r\n    repeat(size) { index ->
list.add(init(index)) }\r\n    return list\r\n}\r\n\r\n/**\r\n * Returns an [IntRange] of the valid indices for this
collection.\r\n * @sample samples.collections.Collections.Collections.indicesOfCollection\r\n */\r\npublic val
Collection<*>.indices: IntRange\r\n    get() = 0..size - 1\r\n\r\n/**\r\n * Returns the index of the last item in the list
or -1 if the list is empty.\r\n * @sample samples.collections.Collections.Lists.lastIndexOfList\r\n */\r\npublic
val <T> List<T>.lastIndex: Int\r\n    get() = this.size - 1\r\n\r\n/**\r\n * Returns `true` if the collection is not

```

```

empty.\r\n * @sample samples.collections.Collections.Collections.collectionIsNotEmpty\r\n
*\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T> Collection<T>.isEmpty(): Boolean =
!isEmpty()\r\n\r\n**\r\n * Returns `true` if this nullable collection is either null or empty.\r\n * @sample
samples.collections.Collections.Collections.collectionIsNullOrEmpty\r\n
*\r\n@SinceKotlin("1.3")\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T>
Collection<T>?.isNullOrEmpty(): Boolean {\r\n    contract {\r\n        returns(false) implies (this@isNullOrEmpty !=
null)\r\n    }\r\n\r\n    return this == null || this.isEmpty()\r\n}\r\n\r\n**\r\n * Returns this Collection if it's not `null`
and the empty list otherwise.\r\n * @sample samples.collections.Collections.Collections.collectionOrElse\r\n
*\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T> Collection<T>?.orElse(): Collection<T> = this ?:
emptyList()\r\n\r\n**\r\n * Returns this List if it's not `null` and the empty list otherwise.\r\n * @sample
samples.collections.Collections.Lists.listOrElse\r\n
*\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T>
List<T>?.orElse(): List<T> = this ?: emptyList()\r\n\r\n**\r\n * Returns this collection if it's not empty\r\n * or
the result of calling [defaultValue] function if the collection is empty.\r\n * \r\n * @sample
samples.collections.Collections.Collections.collectionIfEmpty\r\n
*\r\n@SinceKotlin("1.3")\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <C, R> C.ifEmpty(defaultValue: () -
> R): R where C : Collection<*>, C : R =\r\n    if (isEmpty()) defaultValue() else this\r\n\r\n**\r\n * Checks if
all elements in the specified collection are contained in this collection.\r\n * \r\n * Allows to overcome type-safety
restriction of `containsAll` that requires to pass a collection of type `Collection<E>`.\r\n * @sample
samples.collections.Collections.Collections.collectionContainsAll\r\n
*\r\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in
some cases\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <@kotlin.internal.OnlyInputTypes T>
Collection<T>.containsAll(elements: Collection<T>): Boolean = this.containsAll(elements)\r\n\r\ninternal fun <T>
List<T>.optimizeReadOnlyList() = when (size) {\r\n    0 -> emptyList()\r\n    1 -> listOf(this[0])\r\n    else ->
this\r\n}\r\n\r\n**\r\n * Searches this list or its range for the provided [element] using the binary search
algorithm.\r\n * The list is expected to be sorted into ascending order according to the Comparable natural ordering
of its elements,\r\n * otherwise the result is undefined.\r\n * \r\n * If the list contains multiple elements equal to the
specified [element], there is no guarantee which one will be found.\r\n * \r\n * `null` value is considered to be less
than any non-null value.\r\n * \r\n * @return the index of the element, if it is contained in the list within the specified
range;\r\n * otherwise, the inverted insertion point `(-insertion point - 1)`.\r\n * The insertion point is defined as the
index at which the element should be inserted,\r\n * so that the list (or the specified subrange of list) still remains
sorted.\r\n * @sample samples.collections.Collections.Lists.binarySearchOnComparable\r\n * @sample
samples.collections.Collections.Lists.binarySearchWithBoundaries\r\n
*\r\npublic fun <T : Comparable<T>>
List<T?>.binarySearch(element: T?, fromIndex: Int = 0, toIndex: Int = size): Int {\r\n    rangeCheck(size,
fromIndex, toIndex)\r\n    var low = fromIndex\r\n    var high = toIndex - 1\r\n    while (low <= high) {\r\n
val mid = (low + high).ushr(1) // safe from overflows\r\n        val midVal = get(mid)\r\n        val cmp =
compareValues(midVal, element)\r\n        if (cmp < 0)\r\n            low = mid + 1\r\n        else if (cmp > 0)\r\n
            high = mid - 1\r\n        else\r\n            return mid // key found\r\n    }\r\n    return -(low + 1) // key not
found\r\n}\r\n\r\n**\r\n * Searches this list or its range for the provided [element] using the binary search
algorithm.\r\n * The list is expected to be sorted into ascending order according to the specified [comparator],\r\n *
otherwise the result is undefined.\r\n * \r\n * If the list contains multiple elements equal to the specified [element],
there is no guarantee which one will be found.\r\n * \r\n * `null` value is considered to be less than any non-null
value.\r\n * \r\n * @return the index of the element, if it is contained in the list within the specified range;\r\n *
otherwise, the inverted insertion point `(-insertion point - 1)`.\r\n * The insertion point is defined as the index at
which the element should be inserted,\r\n * so that the list (or the specified subrange of list) still remains sorted
according to the specified [comparator].\r\n * @sample
samples.collections.Collections.Lists.binarySearchWithComparator\r\n
*\r\npublic fun <T>
List<T>.binarySearch(element: T, comparator: Comparator<in T>, fromIndex: Int = 0, toIndex: Int = size): Int {\r\n
rangeCheck(size, fromIndex, toIndex)\r\n    var low = fromIndex\r\n    var high = toIndex - 1\r\n    while

```

```

(low <= high) {
    val mid = (low + high).ushr(1) // safe from overflows
    val midVal = get(mid)
    val cmp = comparator.compare(midVal, element)
    if (cmp < 0) low = mid + 1
    else if (cmp > 0) high = mid - 1
    else return mid // key found
}
return -(low + 1) // key not found
}

/** Searches this list or its range for an element having the key returned by the
specified [selector] function equal to the provided [key] value using the binary search algorithm.
The list is expected to be sorted into ascending order according to the Comparable natural ordering of keys of its
elements.
otherwise the result is undefined.
If the list contains multiple elements with the specified [key], there is no guarantee which one will be found.
`null` value is considered to be less than any non-null value.
@return the index of the element with the specified [key], if it is contained in the list within
the specified range;
otherwise, the inverted insertion point `(-insertion point - 1)`.
The insertion point is defined as the index at which the element should be inserted,
so that the list (or the specified subrange of list) still remains sorted.
@sample samples.collections.Collections.Lists.binarySearchByKey
^
public inline fun <T, K : Comparable<K>> List<T>.binarySearchBy(
    key: K?,
    fromIndex: Int = 0,
    toIndex: Int = size,
    crossinline selector: (T) -> K?): Int =
    binarySearch(fromIndex, toIndex) {
        compareValues(selector(it), key)
    }
// do not introduce this overload --- too rare
public fun <T, K> List<T>.binarySearchBy(key: K, comparator: Comparator<K>, fromIndex: Int = 0, toIndex: Int = size(), selector: (T) -> K): Int =
    binarySearch(fromIndex, toIndex) { comparator.compare(selector(it), key)
}

/** Searches this list or its range for an element for which [comparison] function returns zero using
the binary search algorithm.
The list is expected to be sorted into ascending order according to the provided [comparison],
otherwise the result is undefined.
If the list contains multiple elements for which [comparison] returns zero, there is no guarantee which one will be found.
@param comparison function that compares an element of the list with the element being searched.
@return the index of the found element, if it is contained in the list within the specified range;
otherwise, the inverted insertion point `(-insertion point - 1)`.
The insertion point is defined as the index at which the element should be inserted,
so that the list (or the specified subrange of list) still remains sorted.
@sample samples.collections.Collections.Lists.binarySearchWithComparisonFunction
^
public fun <T> List<T>.binarySearch(fromIndex: Int = 0, toIndex: Int = size, comparison: (T) -> Int): Int {
    rangeCheck(size, fromIndex, toIndex)
    var low = fromIndex
    var high = toIndex - 1
    while (low <= high) {
        val mid = (low + high).ushr(1) // safe from overflows
        val midVal = get(mid)
        val cmp = comparison(midVal)
        if (cmp < 0) low = mid + 1
        else if (cmp > 0) high = mid - 1
        else return mid // key found
    }
    return -(low + 1) // key not found
}

Checks that `from` and `to` are in the range of [0..size] and throws an appropriate exception, if they aren't.
^
private fun rangeCheck(size: Int, fromIndex: Int, toIndex: Int) {
    when {
        fromIndex > toIndex -> throw IllegalArgumentException("`fromIndex ($fromIndex) is greater than toIndex ($toIndex).`)")
        fromIndex < 0 -> throw IndexOutOfBoundsException("`fromIndex ($fromIndex) is less than zero.`")
        toIndex > size -> throw IndexOutOfBoundsException("`toIndex ($toIndex) is greater than size ($size).`)")
    }
}

@PublishedApi
@SinceKotlin("1.3")
internal expect fun checkIndexOverflow(index: Int): Int

@PublishedApi
@SinceKotlin("1.3")
internal expect fun checkCountOverflow(count: Int): Int

@PublishedApi
@SinceKotlin("1.3")
internal fun throwIndexOverflow() { throw ArithmeticException("Index overflow has happened.") }

@PublishedApi
@SinceKotlin("1.3")
internal fun throwCountOverflow() { throw ArithmeticException("Count overflow has happened.") }

/** Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
^
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("MapsKt")
package kotlin.collections
import kotlin.contracts.*
private object EmptyMap : Map<Any?, Nothing>, Serializable {
    private const val serialVersionUID: Long = 8246714829545688274
    override fun equals(other: Any?): Boolean = other is Map<*, *> && other.isEmpty()
    override fun hashCode(): Int = 0
}

```

```

override fun toString(): String = "{}"
override val size: Int get() = 0
override fun isEmpty(): Boolean = true
override fun containsKey(key: Any?): Boolean = false
override fun containsValue(value: Nothing): Boolean = false
override fun get(key: Any?): Nothing? = null
override val entries: Set<Map.Entry<Any?, Nothing>> get() = EmptySet
override val keys: Set<Any?> get() = EmptySet
override val values: Collection<Nothing> get() = EmptyList
private fun readResolve(): Any = EmptyMap
Returns an empty read-only map of specified type.
The returned map is serializable (JVM).
@sample samples.collections.Maps.Instantiation.emptyReadOnlyMap
public fun <K, V> emptyMap(): Map<K, V> = @Suppress("UNCHECKED_CAST") (EmptyMap as Map<K, V>)
Returns a new read-only map with the specified contents, given as a list of pairs where the first value is the key and the second is the value.
If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs.
Entries of the map are iterated in the order they were specified.
The returned map is serializable (JVM).
@sample samples.collections.Maps.Instantiation.mapFromPairs
public fun <K, V> mapOf(vararg pairs: Pair<K, V>): Map<K, V> = if (pairs.size > 0) pairs.toMap(LinkedHashMap(mapCapacity(pairs.size))) else emptyMap()
Returns an empty read-only map.
The returned map is serializable (JVM).
@sample samples.collections.Maps.Instantiation.emptyReadOnlyMap
public fun <K, V> mapOf(): Map<K, V> = emptyMap()
Returns an empty new [MutableMap].
The returned map preserves the entry iteration order.
@sample samples.collections.Maps.Instantiation.emptyMutableMap
public fun <K, V> mutableMapOf(): MutableMap<K, V> = LinkedHashMap()
Returns a new [MutableMap] with the specified contents, given as a list of pairs where the first component is the key and the second is the value.
If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs.
Entries of the map are iterated in the order they were specified.
@sample samples.collections.Maps.Instantiation.mutableMapFromPairs
@sample samples.collections.Maps.Instantiation.emptyMutableMap
public fun <K, V> mutableMapOf(vararg pairs: Pair<K, V>): MutableMap<K, V> = LinkedHashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }
Returns an empty new [HashMap].
@sample samples.collections.Maps.Instantiation.hashMapFromPairs
public fun <K, V> hashMapOf(): HashMap<K, V> = HashMap<K, V>()
Returns a new [HashMap] with the specified contents, given as a list of pairs where the first component is the key and the second is the value.
@sample samples.collections.Maps.Instantiation.hashMapFromPairs
public fun <K, V> hashMapOf(vararg pairs: Pair<K, V>): HashMap<K, V> = HashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }
Returns an empty new [LinkedHashMap].
@sample samples.collections.Maps.Instantiation.linkedMapFromPairs
public fun <K, V> linkedMapOf(): LinkedHashMap<K, V> = LinkedHashMap<K, V>()
Returns a new [LinkedHashMap] with the specified contents, given as a list of pairs where the first component is the key and the second is the value.
If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs.
Entries of the map are iterated in the order they were specified.
@sample samples.collections.Maps.Instantiation.linkedMapFromPairs
public fun <K, V> linkedMapOf(vararg pairs: Pair<K, V>): LinkedHashMap<K, V> = pairs.toMap(LinkedHashMap(mapCapacity(pairs.size)))
Calculate the initial capacity of a map, based on Guava's com.google.common.collect.Maps approach. This is equivalent to the Collection constructor for HashSet, (c.size()/0.75f) + 1, but provides further optimisations for very small or very large sizes, allows support non-collection classes, and provides consistency for all map based class construction.
@sample samples.collections.Maps.Instantiation.mapCapacity
internal fun mapCapacity(expectedSize: Int): Int {
    if (expectedSize < 3) {
        return expectedSize + 1
    }
    if (expectedSize < INT_MAX_POWER_OF_TWO) {
        return expectedSize + expectedSize / 3
    }
    return Int.MAX_VALUE // any large value
}
private const val INT_MAX_POWER_OF_TWO: Int = Int.MAX_VALUE / 2 + 1
Returns `true` if this map is not empty.

```



```

*\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <K, V> Map<out K, V>.isEmpty(): Boolean =
!isEmpty()\r\n\r\n/**\r\n * Returns `true` if this nullable map is either null or empty.\r\n * @sample
samples.collections.Maps.Usage.mapIsNullOrEmpty\r\n
*\r\n@SinceKotlin("1.3")\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <K, V> Map<out K,
V>?.isNullOrEmpty(): Boolean {  

    contract {  

        returns(false) implies (this@isNullOrEmpty != null)\r\n
    }  

    return this == null || isEmpty()\r\n\r\n}\r\n\r\n/**\r\n * Returns the [Map] if its not `null`, or the empty [Map]
otherwise.\r\n * @sample samples.collections.Maps.Usage.mapOrElse\r\n
*\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <K, V> Map<K, V>?.orElse(): Map<K, V> = this ?:
emptyMap()\r\n\r\n/**\r\n * Returns this map if it's not empty\r\n * or the result of calling [defaultValue] function if
the map is empty.\r\n * @sample samples.collections.Maps.Usage.mapIfEmpty\r\n
*\r\n@SinceKotlin("1.3")\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <M, R> M.isEmpty(defaultValue: ()
-> R): R where M : Map<*, *>, M : R =  

    if (isEmpty()) defaultValue() else this\r\n\r\n/**\r\n * Checks if the
map contains the given key.\r\n * This method allows to use the `x in map` syntax for checking whether an
object is contained in the map.\r\n * @kotlin.internal.InlineOnly\r\npublic inline operator fun
<@kotlin.internal.OnlyInputTypes K, V> Map<out K, V>.contains(key: K): Boolean =
containsKey(key)\r\n\r\n/**\r\n * Returns the value corresponding to the given [key], or `null` if such a key is not
present in the map.\r\n * @kotlin.internal.InlineOnly\r\npublic inline operator fun
<@kotlin.internal.OnlyInputTypes K, V> Map<out K, V>.get(key: K): V? =\r\n
@Suppress("UNCHECKED_CAST") (this as Map<K, V>).get(key)\r\n\r\n/**\r\n * Allows to use the index
operator for storing values in a mutable map.\r\n * @kotlin.internal.InlineOnly\r\npublic inline operator fun <K,
V> MutableMap<K, V>.set(key: K, value: V): Unit {  

    put(key, value)\r\n\r\n}\r\n\r\n/**\r\n * Returns `true` if the
map contains the specified [key].\r\n * Allows to overcome type-safety restriction of `containsKey` that
requires to pass a key of type `K`.\r\n * @kotlin.internal.InlineOnly\r\npublic inline fun
<@kotlin.internal.OnlyInputTypes K> Map<out K, *>.containsKey(key: K): Boolean =\r\n
@Suppress("UNCHECKED_CAST") (this as Map<K, *>).containsKey(key)\r\n\r\n/**\r\n * Returns `true` if the
map maps one or more keys to the specified [value].\r\n * Allows to overcome type-safety restriction of
`containsValue` that requires to pass a value of type `V`.\r\n * @sample
samples.collections.Maps.Usage.containsValue\r\n
*\r\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in
some cases\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <K, @kotlin.internal.OnlyInputTypes V> Map<K,
V>.containsValue(value: V): Boolean = this.containsValue(value)\r\n\r\n\r\n/**\r\n * Removes the specified key and
its corresponding value from this map.\r\n * @return the previous value associated with the key, or `null` if the
key was not present in the map.\r\n * Allows to overcome type-safety restriction of `remove` that requires to pass
a key of type `K`.\r\n * @kotlin.internal.InlineOnly\r\npublic inline fun <@kotlin.internal.OnlyInputTypes K,
V> MutableMap<out K, V>.remove(key: K): V? =\r\n    @Suppress("UNCHECKED_CAST") (this as
MutableMap<K, V>).remove(key)\r\n\r\n\r\n/**\r\n * Returns the key component of the map entry.\r\n * This
method allows to use destructuring declarations when working with maps, for example:  

    ```\r\n * for ((key,
value) in map) {

 // do something with the key and the value\r\n * }

    ```\r\n\r\n
*\r\n@kotlin.internal.InlineOnly\r\npublic inline operator fun <K, V> Map.Entry<K, V>.component1(): K =
key\r\n\r\n\r\n/**\r\n * Returns the value component of the map entry.\r\n * This method allows to use
destructuring declarations when working with maps, for example:  

    ```\r\n * for ((key, value) in map) {  

 //
do something with the key and the value\r\n * }

    ```\r\n\r\n}\r\n\r\n\r\n * @kotlin.internal.InlineOnly\r\npublic inline
operator fun <K, V> Map.Entry<K, V>.component2(): V = value\r\n\r\n\r\n\r\n/**\r\n * Converts entry to [Pair] with key
being first component and value being second.\r\n * @kotlin.internal.InlineOnly\r\npublic inline fun <K, V>
Map.Entry<K, V>.toPair(): Pair<K, V> = Pair(key, value)\r\n\r\n\r\n\r\n/**\r\n * Returns the value for the given key, or the
result of the [defaultValue] function if there was no entry for the given key.\r\n * @sample
samples.collections.Maps.Usage.getOrElse\r\n * @kotlin.internal.InlineOnly\r\npublic inline fun <K, V>
Map<K, V>.getOrElse(key: K, defaultValue: () -> V): V = get(key) ?: defaultValue()\r\n\r\n\r\n\r\ninternal inline fun

```

```

<K, V> Map<K, V>.getOrNull(key: K, defaultValue: () -> V): V {
    val value = get(key)
    if (value == null && !containsKey(key)) {
        return defaultValue()
    } else {
        return value
    }
}

@Suppress("UNCHECKED_CAST")
return value as V

Returns the value for the given [key] or throws an exception if there is no such key in the map.
If the map was created by [withDefault], resorts to its `defaultValue` provider function instead of throwing an exception.
throws NoSuchElementException when the map doesn't contain a value for the specified key and no implicit default value was provided for that map.

Since Kotlin("1.1")
public fun <K, V> Map<K, V>.getValue(key: K): V = getOrNull(key)
    ?: getOrDefault(key, defaultValue)

Returns the value for the given key. If the key is not found in the map, calls the [defaultValue] function, puts its result into the map under the given key and returns it.
Note that the operation is not guaranteed to be atomic if the map is being modified concurrently.

sample.samples.collections.maps.usage.getOrPut

public inline fun <K, V> MutableMap<K, V>.getOrPut(key: K, defaultValue: () -> V): V {
    val value = get(key)
    return if (value == null) {
        val answer = defaultValue()
        put(key, answer)
        answer
    } else {
        value
    }
}

Returns an [Iterator] over the entries in the [Map].

sample.samples.collections.maps.usage.forOverEntries

@kotlin.internal.InlineOnly
public inline operator fun <K, V> Map<out K, V>.iterator(): Iterator<Map.Entry<K, V>> = entries.iterator()

Returns a [MutableIterator] over the mutable entries in the [MutableMap].

@kotlin.jvm.JvmName("mutableIterator")
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<K, V>.iterator(): MutableIterator<MutableMap.MutableEntry<K, V>> = entries.iterator()

Populates the given [destination] map with entries having the keys of this map and the values obtained by applying the [transform] function to each entry in this [Map].

public inline fun <K, V, R, M : MutableMap<in K, in R>> Map<out K, V>.mapValuesTo(destination: M, transform: (Map.Entry<K, V>) -> R): M {
    return entries.associateByTo(destination, { it.key }, transform)
}

Populates the given [destination] map with entries having the keys obtained by applying the [transform] function to each entry in this [Map] and the values of this map.
In case if any two entries are mapped to the equal keys, the value of the latter one will overwrite the value associated with the former one.

public inline fun <K, V, R, M : MutableMap<in R, in V>> Map<out K, V>.mapKeysTo(destination: M, transform: (Map.Entry<K, V>) -> R): M {
    return entries.associateByTo(destination, transform, { it.value })
}

Puts all the given [pairs] into this [MutableMap] with the first component in the pair being the key and the second the value.

public fun <K, V> MutableMap<in K, in V>.putAll(pairs: Array<out Pair<K, V>>): Unit {
    for ((key, value) in pairs) {
        put(key, value)
    }
}

Puts all the elements of the given collection into this [MutableMap] with the first component in the pair being the key and the second the value.

public fun <K, V> MutableMap<in K, in V>.putAll(pairs: Iterable<Pair<K, V>>): Unit {
    for ((key, value) in pairs) {
        put(key, value)
    }
}

Puts all the elements of the given sequence into this [MutableMap] with the first component in the pair being the key and the second the value.

public fun <K, V> MutableMap<in K, in V>.putAll(pairs: Sequence<Pair<K, V>>): Unit {
    for ((key, value) in pairs) {
        put(key, value)
    }
}

Returns a new map with entries having the keys of this map and the values obtained by applying the [transform] function to each entry in this [Map].
The returned map preserves the entry iteration order of the original map.

sample.samples.collections.maps.transformations.mapValues

public inline fun <K, V, R> Map<out K, V>.mapValues(transform: (Map.Entry<K, V>) -> R): Map<K, R> {
    return mapValuesTo(LinkedHashMap<K, R>(mapCapacity(size)), transform)
}

.optimizeReadOnlyMap()

Returns a new Map with entries having the keys obtained by applying the [transform] function to each entry in this [Map] and the values of this map.
In case if any two entries are mapped to the equal keys, the value of the latter one will overwrite the value associated with the former one.
The returned map preserves the entry iteration order of the original map.

sample.samples.collections.maps.transformations.mapKeys

public inline fun <K, V, R> Map<out K, V>.mapKeys(transform: (Map.Entry<K, V>) -> R): Map<R, V> {
    return mapKeysTo(LinkedHashMap<R, V>(mapCapacity(size)), transform)
}

Returns a map containing all

```

key-value pairs with keys matching the given [predicate].

```

    * The returned map preserves the entry iteration
    order of the original map.
    * @sample samples.collections.Maps.Filtering.filterKeys
    */
    public inline fun <K, V> Map<out K, V>.filterKeys(predicate: (K) -> Boolean): Map<K, V> {
        val result = LinkedHashMap<K, V>()
        for (entry in this) {
            if (predicate(entry.key)) {
                result.put(entry.key, entry.value)
            }
        }
        return result
    }
    * Returns a map containing all key-value pairs with values matching
    the given [predicate].
    * The returned map preserves the entry iteration order of the original map.
    * @sample samples.collections.Maps.Filtering.filterValues
    */
    public inline fun <K, V> Map<out K,
    V>.filterValues(predicate: (V) -> Boolean): Map<K, V> {
        val result = LinkedHashMap<K, V>()
        for
        (entry in this) {
            if (predicate(entry.value)) {
                result.put(entry.key, entry.value)
            }
        }
        return result
    }
    * Appends all entries matching the given [predicate] into the mutable map given
    as [destination] parameter.
    * @return the destination map.
    * @sample
    samples.collections.Maps.Filtering.filterTo
    */
    public inline fun <K, V, M : MutableMap<in K, in V>>
    Map<out K, V>.filterTo(destination: M, predicate: (Map.Entry<K, V>) -> Boolean): M {
        for (element in this) {
            if (predicate(element)) {
                destination.put(element.key, element.value)
            }
        }
        return destination
    }
    * Returns a new map containing all key-value pairs matching the given
    [predicate].
    * The returned map preserves the entry iteration order of the original map.
    * @sample
    samples.collections.Maps.Filtering.filter
    */
    public inline fun <K, V> Map<out K, V>.filter(predicate:
    (Map.Entry<K, V>) -> Boolean): Map<K, V> {
        return filterTo(LinkedHashMap<K, V>(),
        predicate)
    }
    * Appends all entries not matching the given [predicate] into the given [destination].
    * @return the destination map.
    * @sample samples.collections.Maps.Filtering.filterNotTo
    */
    public inline fun <K, V, M : MutableMap<in K, in V>> Map<out K, V>.filterNotTo(destination: M, predicate:
    (Map.Entry<K, V>) -> Boolean): M {
        for (element in this) {
            if (!predicate(element)) {
                destination.put(element.key, element.value)
            }
        }
        return destination
    }
    * Returns a
    new map containing all key-value pairs not matching the given [predicate].
    * The returned map preserves
    the entry iteration order of the original map.
    * @sample samples.collections.Maps.Filtering.filterNot
    */
    public inline fun <K, V> Map<out K, V>.filterNot(predicate: (Map.Entry<K, V>) -> Boolean): Map<K, V>
    {
        return filterNotTo(LinkedHashMap<K, V>(), predicate)
    }
    * Returns a new map containing
    all key-value pairs from the given collection of pairs.
    * The returned map preserves the entry iteration order
    of the original collection.
    * If any of two pairs would have the same key the last one gets added to the map.
    * @public fun <K, V> Iterable<Pair<K, V>>.toMap(): Map<K, V> {
        if (this is Collection) {
            return
            when (size) {
                0 -> emptyMap()
                1 -> mapOf(if (this is List) this[0] else iterator().next())
            }
            else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
        }
        return
        toMap(LinkedHashMap<K, V>().optimizeReadOnlyMap())
    }
    * Populates and returns the
    [destination] mutable map with key-value pairs from the given collection of pairs.
    * @public fun <K, V, M :
    MutableMap<in K, in V>> Iterable<Pair<K, V>>.toMap(destination: M): M =
        destination.apply {
            putAll(this@toMap)
        }
    * Returns a new map containing all key-value pairs from the given array of
    pairs.
    * The returned map preserves the entry iteration order of the original array.
    * If any of two pairs
    would have the same key the last one gets added to the map.
    * @public fun <K, V> Array<out Pair<K,
    V>>.toMap(): Map<K, V> = when (size) {
        0 -> emptyMap()
        1 -> mapOf(this[0])
        else ->
        toMap(LinkedHashMap<K, V>(mapCapacity(size)))
    }
    * Populates and returns the [destination]
    mutable map with key-value pairs from the given array of pairs.
    * @public fun <K, V, M : MutableMap<in K,
    in V>> Array<out Pair<K, V>>.toMap(destination: M): M =
        destination.apply {
            putAll(this@toMap)
        }
    * Returns a new map containing all key-value pairs from the given sequence of pairs.
    * The
    returned map preserves the entry iteration order of the original sequence.
    * If any of two pairs would have the
    same key the last one gets added to the map.
    * @public fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K,
    V> = toMap(LinkedHashMap<K, V>().optimizeReadOnlyMap())
    * Populates and returns the
    [destination] mutable map with key-value pairs from the given sequence of pairs.
    * @public fun <K, V, M :
    MutableMap<in K, in V>> Sequence<Pair<K, V>>.toMap(destination: M): M =
        destination.apply {

```

`putAll(this@toMap)` } Returns a new read-only map containing all key-value pairs from the original map.
 * The returned map preserves the entry iteration order of the original map.
`*/@SinceKotlin("1.1")\npublic fun <K, V> Map<out K, V>.toMap(): Map<K, V> = when (size) { 0 -> emptyMap() 1 -> toSingletonMap() else -> toMutableMap() }`
 * Returns a new mutable map containing all key-value pairs from the original map.
 * The returned map preserves the entry iteration order of the original map.
`*/@SinceKotlin("1.1")\npublic fun <K, V> Map<out K, V>.toMutableMap(): MutableMap<K, V> = LinkedHashMap(this)`
 * Populates and returns the [destination] mutable map with key-value pairs from the given map.
`*/@SinceKotlin("1.1")\npublic fun <K, V, M : MutableMap<in K, in V>> Map<out K, V>.toMap(destination: M): M = destination.apply { putAll(this@toMap) }`
 * Creates a new read-only map by replacing or adding an entry to this map from a given key-value [pair].
 * The returned map preserves the entry iteration order of the original map.
 * The [pair] is iterated in the end if it has a unique key.
`*/\npublic operator fun <K, V> Map<out K, V>.plus(pair: Pair<K, V>): Map<K, V> = if (this.isEmpty()) mapOf(pair) else LinkedHashMap(this).apply { put(pair.first, pair.second) }`
 * Creates a new read-only map by replacing or adding entries to this map from a given collection of key-value [pairs].
 * The returned map preserves the entry iteration order of the original map.
 * Those [pairs] with unique keys are iterated in the end in the order of [pairs] collection.
`*/\npublic operator fun <K, V> Map<out K, V>.plus(pairs: Iterable<Pair<K, V>>): Map<K, V> = if (this.isEmpty()) pairs.toMap() else LinkedHashMap(this).apply { putAll(pairs) }`
 * Creates a new read-only map by replacing or adding entries to this map from a given array of key-value [pairs].
 * The returned map preserves the entry iteration order of the original map.
 * Those [pairs] with unique keys are iterated in the end in the order of [pairs] array.
`*/\npublic operator fun <K, V> Map<out K, V>.plus(pairs: Array<out Pair<K, V>>): Map<K, V> = if (this.isEmpty()) pairs.toMap() else LinkedHashMap(this).apply { putAll(pairs) }`
 * Creates a new read-only map by replacing or adding entries to this map from a given sequence of key-value [pairs].
 * The returned map preserves the entry iteration order of the original map.
 * Those [pairs] with unique keys are iterated in the end in the order of [pairs] sequence.
`*/\npublic operator fun <K, V> Map<out K, V>.plus(pairs: Sequence<Pair<K, V>>): Map<K, V> = LinkedHashMap(this).apply { putAll(pairs) }.optimizeReadOnlyMap()`
 * Creates a new read-only map by replacing or adding entries to this map from another [map].
 * The returned map preserves the entry iteration order of the original map.
 * Those entries of another [map] that are missing in this map are iterated in the end in the order of that [map].
`*/\npublic operator fun <K, V> Map<out K, V>.plus(map: Map<out K, V>): Map<K, V> = LinkedHashMap(this).apply { putAll(map) }`
 * Appends or replaces the given [pair] in this mutable map.
`*/@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pair: Pair<K, V>) { put(pair.first, pair.second) }`
 * Appends or replaces all pairs from the given collection of [pairs] in this mutable map.
`*/@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Iterable<Pair<K, V>>) { putAll(pairs) }`
 * Appends or replaces all pairs from the given array of [pairs] in this mutable map.
`*/@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Array<out Pair<K, V>>) { putAll(pairs) }`
 * Appends or replaces all pairs from the given sequence of [pairs] in this mutable map.
`*/@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Sequence<Pair<K, V>>) { putAll(pairs) }`
 * Appends or replaces all entries from the given [map] in this mutable map.
`*/@kotlin.internal.InlineOnly\npublic inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(map: Map<K, V>) { putAll(map) }`
 * Returns a map containing all entries of the original map except the entry with the given [key].
 * The returned map preserves the entry iteration order of the original map.
`*/@SinceKotlin("1.1")\npublic operator fun <K, V> Map<out K, V>.minus(key: K): Map<K, V> = this.toMutableMap().apply { minusAssign(key) }.optimizeReadOnlyMap()`
 * Returns a map containing all entries of the original map except those entries the keys of which are contained in the given [keys] collection.
 * The returned map preserves the entry iteration order of the original map.

```

*\/r\n@SinceKotlin("1.1")r\npublic operator fun <K, V> Map<out K, V>.minus(keys: Iterable<K>): Map<K, V>
=r\n this.toMutableMap().apply { minusAssign(keys) }.optimizeReadOnlyMap()\r\n\r\n/**r\n * Returns a map
containing all entries of the original map except those entries\r\n * the keys of which are contained in the given
[keys] array.\r\n *r\n * The returned map preserves the entry iteration order of the original map.\r\n
*\/r\n@SinceKotlin("1.1")r\npublic operator fun <K, V> Map<out K, V>.minus(keys: Array<out K>): Map<K,
V> =r\n this.toMutableMap().apply { minusAssign(keys) }.optimizeReadOnlyMap()\r\n\r\n/**r\n * Returns a
map containing all entries of the original map except those entries\r\n * the keys of which are contained in the given
[keys] sequence.\r\n *r\n * The returned map preserves the entry iteration order of the original map.\r\n
*\/r\n@SinceKotlin("1.1")r\npublic operator fun <K, V> Map<out K, V>.minus(keys: Sequence<K>): Map<K,
V> =r\n this.toMutableMap().apply { minusAssign(keys) }.optimizeReadOnlyMap()\r\n\r\n/**r\n * Removes the
entry with the given [key] from this mutable map.\r\n
*\/r\n@SinceKotlin("1.1")r\n@kotlin.internal.InlineOnlyr\npublic inline operator fun <K, V> MutableMap<K,
V>.minusAssign(key: K) {\r\n remove(key)\r\n}\r\n\r\n/**r\n * Removes all entries the keys of which are
contained in the given [keys] collection from this mutable map.\r\n
*\/r\n@SinceKotlin("1.1")r\n@kotlin.internal.InlineOnlyr\npublic inline operator fun <K, V> MutableMap<K,
V>.minusAssign(keys: Iterable<K>) {\r\n this.keys.removeAll(keys)\r\n}\r\n\r\n/**r\n * Removes all entries the
keys of which are contained in the given [keys] array from this mutable map.\r\n
*\/r\n@SinceKotlin("1.1")r\n@kotlin.internal.InlineOnlyr\npublic inline operator fun <K, V> MutableMap<K,
V>.minusAssign(keys: Array<out K>) {\r\n this.keys.removeAll(keys)\r\n}\r\n\r\n/**r\n * Removes all entries
from the keys of which are contained in the given [keys] sequence from this mutable map.\r\n
*\/r\n@SinceKotlin("1.1")r\n@kotlin.internal.InlineOnlyr\npublic inline operator fun <K, V> MutableMap<K,
V>.minusAssign(keys: Sequence<K>) {\r\n this.keys.removeAll(keys)\r\n}\r\n\r\n\r\n// do not expose for now
@PublishedApi\r\ninternal fun <K, V> Map<K, V>.optimizeReadOnlyMap() = when (size) {\r\n 0 ->
emptyMap()\r\n 1 -> toSingletonMapOrSelf()\r\n else -> this\r\n}\r\n", "*/r\n * Copyright 2010-2018 JetBrains
s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0
license that can be found in the license/LICENSE.txt file.\r\n
*\/r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("SetsKt")\r\n\r\npackage
kotlin.collections\r\n\r\n\r\ninternal object EmptySet : Set<Nothing>, Serializable {\r\n private const val
serialVersionUID: Long = 3406603774387020532\r\n\r\n override fun equals(other: Any?): Boolean = other is
Set<*> && other.isEmpty()\r\n override fun hashCode(): Int = 0\r\n override fun toString(): String =
"[]"\r\n\r\n override val size: Int get() = 0\r\n override fun isEmpty(): Boolean = true\r\n override fun
contains(element: Nothing): Boolean = false\r\n override fun containsAll(elements: Collection<Nothing>):
Boolean = elements.isEmpty()\r\n\r\n override fun iterator(): Iterator<Nothing> = EmptyIterator\r\n\r\n private
fun readResolve(): Any = EmptySet\r\n}\r\n\r\n\r\n/**r\n * Returns an empty read-only set. The returned set is
serializable (JVM).\r\n * @sample samples.collections.Collections.Sets.emptyReadOnlySet\r\n *\/r\npublic fun <T>
emptySet(): Set<T> = EmptySet\r\n\r\n\r\n/**r\n * Returns a new read-only set with the given elements.\r\n * Elements
of the set are iterated in the order they were specified.\r\n * The returned set is serializable (JVM).\r\n * @sample
samples.collections.Collections.Sets.readOnlySet\r\n *\/r\npublic fun <T> setOf(vararg elements: T): Set<T> = if
(elements.size > 0) elements.toSet() else emptySet()\r\n\r\n\r\n/**r\n * Returns an empty read-only set. The returned
set is serializable (JVM).\r\n * @sample samples.collections.Collections.Sets.emptyReadOnlySet\r\n
*\/r\n@kotlin.internal.InlineOnlyr\npublic inline fun <T> setOf(): Set<T> = emptySet()\r\n\r\n\r\n/**r\n * Returns an
empty new [MutableSet].\r\n *r\n * The returned set preserves the element iteration order.\r\n * @sample
samples.collections.Collections.Sets.emptyMutableSet\r\n
*\/r\n@SinceKotlin("1.1")r\n@kotlin.internal.InlineOnlyr\npublic inline fun <T> mutableSetOf():
MutableSet<T> = LinkedHashSet()\r\n\r\n\r\n/**r\n * Returns a new [MutableSet] with the given elements.\r\n *
Elements of the set are iterated in the order they were specified.\r\n * @sample
samples.collections.Collections.Sets.mutableSet\r\n *\/r\npublic fun <T> mutableSetOf(vararg elements: T):
MutableSet<T> = elements.toCollection(LinkedHashSet(mapCapacity(elements.size)))\r\n\r\n\r\n/** Returns an empty

```

```

new [HashSet]. *@SinceKotlin("1.1")\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T> hashSetOf():
HashSet<T> = HashSet()\r\n\r\n/** Returns a new [HashSet] with the given elements. */\r\npublic fun <T>
hashSetOf(vararg elements: T): HashSet<T> =
elements.toCollection(HashSet(mapCapacity(elements.size)))\r\n\r\n/** Returns an empty new [LinkedHashSet].
*/\r\n@SinceKotlin("1.1")\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T> linkedSetOf():
LinkedHashSet<T> = LinkedHashSet()\r\n\r\n/**\r\n * Returns a new [LinkedHashSet] with the given elements.\r\n
* Elements of the set are iterated in the order they were specified.\r\n
*/\r\npublic fun <T> linkedSetOf(vararg
elements: T): LinkedHashSet<T> = elements.toCollection(LinkedHashSet(mapCapacity(elements.size)))\r\n\r\n\r\n/**
Returns this Set if it's not `null` and the empty set otherwise. */\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
<T> Set<T>?.orEmpty(): Set<T> = this ?: emptySet()\r\n\r\ninternal fun <T> Set<T>.optimizeReadOnlySet() =
when (size) {\r\n    0 -> emptySet()\r\n    1 -> setOf(iterator().next())\r\n    else -> this\r\n}\r\n\r\n", /*\r\n * Copyright
2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n
* Use of this source code is
governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*/\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("StringsKt")\r\n@file:Suppress("PLA
TFORM_CLASS_MAPPED_TO_KOTLIN")\r\n\r\npackage kotlin.text\r\n\r\n/**\r\n * Parses the string as a signed
[Byte] number and returns the result\r\n * or `null` if the string is not a valid representation of a number.\r\n
*/\r\n@SinceKotlin("1.1")\r\npublic fun String.toByteArrayOrNull(): Byte? = toByteOrNull(radix = 10)\r\n\r\n\r\n/**\r\n *
Parses the string as a signed [Byte] number and returns the result\r\n * or `null` if the string is not a valid
representation of a number.\r\n
* @throws IllegalArgumentException when [radix] is not a valid radix for
string to number conversion.\r\n
*/\r\n@SinceKotlin("1.1")\r\npublic fun String.toByteArrayOrNull(radix: Int): Byte?
{\r\n    val int = this.toIntOrNull(radix) ?: return null\r\n    if (int < Byte.MIN_VALUE || int > Byte.MAX_VALUE)
return null\r\n    return int.toByteArray()\r\n}\r\n\r\n\r\n/**\r\n * Parses the string as a [Short] number and returns the
result\r\n * or `null` if the string is not a valid representation of a number.\r\n
*/\r\n@SinceKotlin("1.1")\r\npublic
fun String.toShortOrNull(): Short? = toShortOrNull(radix = 10)\r\n\r\n\r\n/**\r\n * Parses the string as a [Short] number
and returns the result\r\n * or `null` if the string is not a valid representation of a number.\r\n
* @throws
IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\r\n
*/\r\n@SinceKotlin("1.1")\r\npublic fun String.toShortOrNull(radix: Int): Short? {\r\n    val int =
this.toIntOrNull(radix) ?: return null\r\n    if (int < Short.MIN_VALUE || int > Short.MAX_VALUE) return null\r\n
return int.toShort()\r\n}\r\n\r\n\r\n/**\r\n * Parses the string as an [Int] number and returns the result\r\n * or `null` if
the string is not a valid representation of a number.\r\n
*/\r\n@SinceKotlin("1.1")\r\npublic fun
String.toIntOrNull(): Int? = toIntOrNull(radix = 10)\r\n\r\n\r\n/**\r\n * Parses the string as an [Int] number and returns
the result\r\n * or `null` if the string is not a valid representation of a number.\r\n
* @throws
IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\r\n
*/\r\n@SinceKotlin("1.1")\r\npublic fun String.toIntOrNull(radix: Int): Int? {\r\n    checkRadix(radix)\r\n\r\n    val
length = this.length\r\n    if (length == 0) return null\r\n\r\n    val start: Int\r\n    val isNegative: Boolean\r\n    val
limit: Int\r\n\r\n    val firstChar = this[0]\r\n    if (firstChar < '0') { // Possible leading sign\r\n        if (length == 1)
return null // non-digit (possible sign) only, no digits after\r\n\r\n        start = 1\r\n\r\n        if (firstChar == '-') {\r\n
            isNegative = true\r\n            limit = Int.MIN_VALUE\r\n        } else if (firstChar == '+') {\r\n
            isNegative = false\r\n            limit = -Int.MAX_VALUE\r\n        } else\r\n            return null\r\n        } else {\r\n
            start = 0\r\n            isNegative = false\r\n            limit = -Int.MAX_VALUE\r\n        }\r\n\r\n\r\n    val limitBeforeMul = limit / radix\r\n    var
result = 0\r\n    for (i in start..(length - 1)) {\r\n        val digit = digitOf(this[i], radix)\r\n\r\n        if (digit < 0) return
null\r\n        if (result < limitBeforeMul) return null\r\n\r\n        result *= radix\r\n\r\n        if (result < limit + digit)
return null\r\n\r\n        result -= digit\r\n    }\r\n\r\n    return if (isNegative) result else -result\r\n}\r\n\r\n\r\n/**\r\n *
Parses the string as a [Long] number and returns the result\r\n * or `null` if the string is not a valid representation of
a number.\r\n
*/\r\n@SinceKotlin("1.1")\r\npublic fun String.toLongOrNull(): Long? = toLongOrNull(radix =
10)\r\n\r\n\r\n/**\r\n * Parses the string as a [Long] number and returns the result\r\n * or `null` if the string is not a
valid representation of a number.\r\n
* @throws IllegalArgumentException when [radix] is not a valid radix
for string to number conversion.\r\n
*/\r\n@SinceKotlin("1.1")\r\npublic fun String.toLongOrNull(radix: Int):

```

```

Long? {
    checkRadix(radix)
    val length = this.length
    if (length == 0) return null
    val start:
    Int
    val isNegative: Boolean
    val limit: Long
    val firstChar = this[0]
    if (firstChar < '0') { //
    Possible leading sign
        if (length == 1) return null // non-digit (possible sign) only, no digits after
        start = 1
        if (firstChar == '-') {
            isNegative = true
            limit = Long.MIN_VALUE
        } else if (firstChar == '+') {
            isNegative = false
            limit = -Long.MAX_VALUE
        } else
            return null
        } else {
            start = 0
            isNegative = false
            limit = -Long.MAX_VALUE
        }
    }
    val limitBeforeMul = limit / radix
    var result = 0L
    for (i in start..(length - 1)) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result < limitBeforeMul) return
        null
        result *= radix
        if (result < limit + digit) return null
        result -= digit
    }
    return if (isNegative) result else -result
}
internal fun numberFormatError(input: String):
Nothing = throw NumberFormatException("Invalid number format: '$input'")
/*
 * Copyright 2010-2018
 JetBrains s.r.o. and Kotlin Programming Language contributors.
 Use of this source code is governed by the
 Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
@file:kotlin.jvm.JvmName("UnsignedKt")
@file:UseExperimental(ExperimentalUnsignedTypes::class)
package kotlin
@PublishedApi
internal fun uintCompare(v1: Int, v2: Int): Int = (v1 xor
Int.MIN_VALUE).compareTo(v2 xor Int.MIN_VALUE)
@PublishedApi
internal fun ulongCompare(v1:
Long, v2: Long): Int = (v1 xor Long.MIN_VALUE).compareTo(v2 xor
Long.MIN_VALUE)
@PublishedApi
internal fun uintDivide(v1: UInt, v2: UInt): UInt = (v1.toLong() /
v2.toLong()).toUInt()
@PublishedApi
internal fun uintRemainder(v1: UInt, v2: UInt): UInt = (v1.toLong() %
v2.toLong()).toUInt()
// Division and remainder are based on Guava's UnsignedLongs implementation
// Copyright 2011 The Guava Authors
@PublishedApi
internal fun ulongDivide(v1: ULong, v2: ULong):
ULong {
    val dividend = v1.toLong()
    val divisor = v2.toLong()
    if (divisor < 0) { // i.e., divisor >=
2^63
        return if (v1 < v2) ULong(0) else ULong(1)
    }
    // Optimization - use signed division if
both dividend and divisor < 2^63
    if (dividend >= 0) {
        return ULong(dividend / divisor)
    }
    // Otherwise, approximate the quotient, check, and correct if necessary.
    val quotient = ((dividend ushr 1) /
divisor) shl 1
    val rem = dividend - quotient * divisor
    return ULong(quotient + if (ULong(rem) >=
ULong(divisor)) 1 else 0)
}
@PublishedApi
internal fun ulongRemainder(v1: ULong, v2: ULong):
ULong {
    val dividend = v1.toLong()
    val divisor = v2.toLong()
    if (divisor < 0) { // i.e., divisor >=
2^63
        return if (v1 < v2) {
            v1 // dividend < divisor
        } else {
            v1 - v2 // dividend
>= divisor
        }
    }
    // Optimization - use signed modulus if both dividend and divisor < 2^63
    if (dividend >= 0) {
        return ULong(dividend % divisor)
    }
    // Otherwise, approximate the
quotient, check, and correct if necessary.
    val quotient = ((dividend ushr 1) / divisor) shl 1
    val rem =
dividend - quotient * divisor
    return ULong(rem - if (ULong(rem) >= ULong(divisor)) divisor else
0)
}
@PublishedApi
internal fun doubleToUInt(v: Double): UInt = when {
    v.isNaN() -> 0u
    v <= UInt.MIN_VALUE.toDouble() -> UInt.MIN_VALUE
    v >= UInt.MAX_VALUE.toDouble() ->
UInt.MAX_VALUE
    v <= Int.MAX_VALUE -> v.toInt().toUInt()
    else -> (v -
Int.MAX_VALUE).toInt().toUInt() + Int.MAX_VALUE.toUInt() // Int.MAX_VALUE < v <
UInt.MAX_VALUE
}
@PublishedApi
internal fun doubleToULong(v: Double): ULong = when {
    v.isNaN() -> 0u
    v <= ULong.MIN_VALUE.toDouble() -> ULong.MIN_VALUE
    v >=
ULong.MAX_VALUE.toDouble() -> ULong.MAX_VALUE
    v < Long.MAX_VALUE ->
v.toLong().toULong()
    // Real values from Long.MAX_VALUE to (Long.MAX_VALUE + 1) are not
representable in Double, so don't handle them.
    else -> (v - 9223372036854775808.0).toLong().toULong() +
9223372036854775808uL // Long.MAX_VALUE + 1 < v <
ULong.MAX_VALUE
}
@PublishedApi
internal fun uintToDouble(v: Int): Double = (v and
Int.MAX_VALUE).toDouble() + (v ushr 31 shl 30).toDouble() * 2
@PublishedApi
internal fun
ulongToDouble(v: Long): Double = (v ushr 11).toDouble() * 2048 + (v and 2047)
@PublishedApi
internal fun
ulongToString(v: Long): String = ulongToString(v, 10)
@PublishedApi
internal fun ulongToString(v: Long, base: Int): String
{
    if (v >= 0) return v.toString(base)
    var quotient = ((v ushr 1) / base) shl 1
    var rem = v - quotient
}

```

```

 * base\r\n if (rem >= base) {\r\n     rem -= base\r\n     quotient += 1\r\n } \r\n return quotient.toString(base)
 + rem.toString(base)\r\n}\r\n\r\n"/ * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
 license/LICENSE.txt file.\r\n
 */
@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("CollectionsKt")\r\n\r\npackage
 kotlin.collections\r\n\r\n/*\r\n * Given an [iterator] function constructs an [Iterable] instance that returns values
 through the [Iterator]\r\n * provided by that function.\r\n * @sample
 samples.collections.Iterables.Building.iterable\r\n *
 */
@kotlin.internal.InlineOnly\r\npublic inline fun <T>
 Iterable(crossinline iterator: () -> Iterator<T>): Iterable<T> = object : Iterable<T> {\r\n     override fun iterator():
 Iterator<T> = iterator()\r\n}\r\n\r\n/*\r\n * A wrapper over another [Iterable] (or any other object that can produce
 an [Iterator]) that returns\r\n * an indexing iterator.\r\n *
 */
@kotlin.internal class IndexingIterable<out T>(private val
 iteratorFactory: () -> Iterator<T>) : Iterable<IndexedValue<T>> {\r\n     override fun iterator():
 Iterator<IndexedValue<T>> = IndexingIterator(iteratorFactory())\r\n}\r\n\r\n\r\n/*\r\n * Returns the size of this
 iterable if it is known, or `null` otherwise.\r\n *
 */
@PublishedApi\r\ninternal fun <T>
 Iterable<T>.collectionSizeOrNull(): Int? = if (this is Collection<*>) this.size else null\r\n\r\n/*\r\n * Returns the
 size of this iterable if it is known, or the specified [default] value otherwise.\r\n *
 */
@PublishedApi\r\ninternal fun
 <T> Iterable<T>.collectionSizeOrDefault(default: Int): Int = if (this is Collection<*>) this.size else
 default\r\n\r\n/*\r\n * Returns true when it's safe to convert this collection to a set without changing contains method
 behavior.
 */
private fun <T> Collection<T>.safeToConvertToSet() = size > 2 && this is ArrayList\r\n\r\n/*\r\n * Converts this collection to a set, when it's worth so and it doesn't change contains method behavior.
 */
internal fun
 <T> Iterable<T>.convertToSetForSetOperationWith(source: Iterable<T>): Collection<T> =\r\n     when (this)
 {\r\n         is Set -> this\r\n         is Collection ->\r\n             when {\r\n                 source is Collection && source.size <
 2 -> this\r\n                 else -> if (this.safeToConvertToSet()) toHashSet() else this\r\n             }\r\n         else ->
 toHashSet()\r\n     }\r\n\r\n/*\r\n * Converts this collection to a set, when it's worth so and it doesn't change contains
 method behavior.
 */
internal fun <T> Iterable<T>.convertToSetForSetOperation(): Collection<T> =\r\n     when
 (this) {\r\n         is Set -> this\r\n         is Collection -> if (this.safeToConvertToSet()) toHashSet() else this\r\n
     else -> toHashSet()\r\n     }\r\n\r\n\r\n/*\r\n * Returns a single list of all elements from all collections in the given
 collection.\r\n * @sample samples.collections.Iterables.Operations.flattenIterable\r\n *
 */
public fun <T>
 Iterable<Iterable<T>>.flatten(): List<T> {\r\n     val result = ArrayList<T>()\r\n     for (element in this) {\r\n
 result.addAll(element)\r\n     }\r\n     return result\r\n}\r\n\r\n\r\n/*\r\n * Returns a pair of lists, where\r\n * *first* list is
 built from the first values of each pair from this collection,\r\n * *second* list is built from the second values of each
 pair from this collection.\r\n * @sample samples.collections.Iterables.Operations.unzipIterable\r\n *
 */
public fun
 <T, R> Iterable<Pair<T, R>>.unzip(): Pair<List<T>, List<R>> {\r\n     val expectedSize =
 collectionSizeOrDefault(10)\r\n     val listT = ArrayList<T>(expectedSize)\r\n     val listR =
 ArrayList<R>(expectedSize)\r\n     for (pair in this) {\r\n         listT.add(pair.first)\r\n         listR.add(pair.second)\r\n
     }\r\n     return listT to listR\r\n}\r\n\r\n"/ * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
 Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
 license/LICENSE.txt file.\r\n
 */
@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("SequencesKt")\r\n\r\npackage
 kotlin.sequences\r\n\r\n/*\r\n * Given an [iterator] function constructs a [Sequence] that returns values through the
 [Iterator]\r\n * provided by that function.\r\n * The values are evaluated lazily, and the sequence is potentially
 infinite.\r\n *
 */
\r\n * @sample samples.collections.Sequences.Building.sequenceFromIterator\r\n
 */
@kotlin.internal.InlineOnly\r\npublic inline fun <T> Sequence(crossinline iterator: () -> Iterator<T>):
 Sequence<T> = object : Sequence<T> {\r\n     override fun iterator(): Iterator<T> = iterator()\r\n}\r\n\r\n\r\n/*\r\n * Creates a sequence that returns all elements from this iterator. The sequence is constrained to be iterated only
 once.\r\n * @sample samples.collections.Sequences.Building.sequenceFromIterator\r\n *
 */
public fun <T>
 Iterator<T>.asSequence(): Sequence<T> = Sequence { this }.constrainOnce()\r\n\r\n\r\n/*\r\n * Creates a sequence
 that returns the specified values.\r\n * @sample samples.collections.Sequences.Building.sequenceOfValues\r\n
 */

```



```

*public fun <T> sequenceOf(vararg elements: T): Sequence<T> = if (elements.isEmpty()) emptySequence()
else elements.asSequence()
* Returns an empty sequence.
public fun <T> emptySequence():
Sequence<T> = EmptySequence
private object EmptySequence : Sequence<Nothing>,
DropTakeSequence<Nothing> {
    override fun iterator(): Iterator<Nothing> = EmptyIterator
    override fun
drop(n: Int) = EmptySequence
    override fun take(n: Int) = EmptySequence
}
* Returns this
sequence if it's not `null` and the empty sequence otherwise.
* @sample
samples.collections.Sequences.Usage.sequenceOrEmpty

*@SinceKotlin("1.3")@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>?.orEmpty():
Sequence<T> = this ?: emptySequence()
* Returns a sequence that iterates through the elements
either of this sequence or, if this sequence turns out to be empty, of the sequence returned by [defaultValue]
function.
* @sample samples.collections.Sequences.Usage.sequenceIfEmpty

*@SinceKotlin("1.3")
public fun <T> Sequence<T>.ifEmpty(defaultValue: () -> Sequence<T>):
Sequence<T> = sequence {
    val iterator = this@ifEmpty.iterator()
    if (iterator.hasNext()) {
        yieldAll(iterator)
    } else {
        yieldAll(defaultValue())
    }
}
* Returns a sequence of all
elements from all sequences in this sequence.
* The operation is _intermediate_ and _stateless_.

public fun <T> Sequence<Sequence<T>>.flatten(): Sequence<T> = flatten { it.iterator() }
* Returns a sequence of all elements from all iterables in this sequence.
* The operation is _intermediate_ and
_stateless_.
@kotlin.jvm.JvmName("flattenSequenceOfIterable")
public fun <T>
Sequence<Iterable<T>>.flatten(): Sequence<T> = flatten { it.iterator() }
private fun <T, R>
Sequence<T>.flatten(iterator: (T) -> Iterator<R>): Sequence<R> {
    if (this is TransformingSequence<*, *>) {
        return (this as TransformingSequence<*, T>).flatten(iterator)
    }
    return FlatteningSequence(this,
{ it }, iterator)
}
* Returns a pair of lists, where *first* list is built from the first values of each
pair from this sequence, *second* list is built from the second values of each pair from this sequence.
* The operation is _terminal_.

public fun <T, R> Sequence<Pair<T, R>>.unzip(): Pair<List<T>, List<R>>
{
    val listT = ArrayList<T>()
    val listR = ArrayList<R>()
    for (pair in this) {
        listT.add(pair.first)
        listR.add(pair.second)
    }
    return listT to listR
}
* A sequence
that returns the values from the underlying [sequence] that either match or do not match
the specified
[predicate].
* @param sendWhen If `true`, values for which the predicate returns `true` are returned.
Otherwise, values for which the predicate returns `false` are returned.

internal class
FilteringSequence<T>(\r\n private val sequence: Sequence<T>, \r\n private val sendWhen: Boolean = true, \r\n
private val predicate: (T) -> Boolean) : Sequence<T> {
    override fun iterator(): Iterator<T> = object :
Iterator<T> {
        val iterator = sequence.iterator()
        var nextState: Int = -1 // -1 for unknown, 0 for done, 1
for continue
        var nextItem: T? = null
        private fun calcNext() {
            while (iterator.hasNext()) {
                val item = iterator.next()
                if (predicate(item) == sendWhen) {
                    nextItem =
item
                    nextState = 1
                    return
                }
                nextState = 0
            }
        }
        override fun next(): T {
            if (nextState == -1) {
                calcNext()
                if (nextState
== 0) {
                    throw NoSuchElementException()
                }
                val result = nextItem
                nextItem = null
                nextState = -1
                @SuppressWarnings("UNCHECKED_CAST")
                return result as T
            }
            override fun hasNext(): Boolean {
                if (nextState == -1) {
                    calcNext()
                    return nextState
== 1
                }
            }
        }
}
* A sequence which returns the results of applying the given [transformer]
function to the values
in the underlying [sequence].

internal class TransformingSequence<T,
R>(\r\n constructor(private val sequence: Sequence<T>, private val transformer: (T) -> R) : Sequence<R> {
    override fun iterator(): Iterator<R> = object : Iterator<R> {
        val iterator = sequence.iterator()
        override
fun next(): R {
            return transformer(iterator.next())
        }
        override fun hasNext(): Boolean
{
            return iterator.hasNext()
        }
    }
    internal fun <E> flatten(iterator: (R) -> Iterator<E>):
Sequence<E> {
        return FlatteningSequence<T, R, E>(sequence, transformer, iterator)
    }
}
* A sequence which returns the results of applying the given [transformer] function to the
values
in the underlying [sequence], where the transformer function takes the index of the value in the

```

```

underlying * sequence along with the value itself.
internal class TransformingIndexedSequence<T,
R> constructor(private val sequence: Sequence<T>, private val transformer: (Int, T) -> R) : Sequence<R> {
  override fun iterator(): Iterator<R> = object : Iterator<R> {
    val iterator = sequence.iterator()
    var index = 0
    override fun next(): R {
      return transformer(checkIndexOverflow(index++),
iterator.next())
    }
    override fun hasNext(): Boolean {
      return iterator.hasNext()
    }
  }
}
internal class A sequence which combines values from the underlying [sequence] with their indices
and returns them as [IndexedValue] objects.
internal class IndexingSequence<T> constructor(private val sequence: Sequence<T>) : Sequence<IndexedValue<T>> {
  override fun iterator(): Iterator<IndexedValue<T>> = object : Iterator<IndexedValue<T>> {
    val iterator =
sequence.iterator()
    var index = 0
    override fun next(): IndexedValue<T> {
      return
IndexedValue(checkIndexOverflow(index++), iterator.next())
    }
    override fun hasNext(): Boolean {
      return iterator.hasNext()
    }
  }
}
internal class MergingSequence<T1, T2, V> constructor(
private val sequence1:
Sequence<T1>,
private val sequence2: Sequence<T2>,
private val transform: (T1, T2) -> V) :
Sequence<V> {
  override fun iterator(): Iterator<V> = object : Iterator<V> {
    val iterator1 =
sequence1.iterator()
    val iterator2 = sequence2.iterator()
    override fun next(): V {
      return
transform(iterator1.next(), iterator2.next())
    }
    override fun hasNext(): Boolean {
      return
iterator1.hasNext() && iterator2.hasNext()
    }
  }
}
internal class FlatteningSequence<T, R,
E> constructor(
private val sequence: Sequence<T>,
private val transformer: (T) -> R,
private val
iterator: (R) -> Iterator<E>) : Sequence<E> {
  override fun iterator(): Iterator<E> = object : Iterator<E> {
    val iterator = sequence.iterator()
    var itemIterator: Iterator<E>? = null
    override fun
next(): E {
      if (!ensureItemIterator())
throw NoSuchElementException()
      return
itemIterator!!.next()
    }
    override fun hasNext(): Boolean {
      return
ensureItemIterator()
    }
    private fun ensureItemIterator(): Boolean {
      if
(itemIterator?.hasNext() == false)
itemIterator = null
      while (itemIterator == null) {
        if (iterator.hasNext()) {
          return false
        } else {
          val element =
iterator.next()
          val nextItemIterator = iterator(transformer(element))
          if
(nextItemIterator.hasNext())
itemIterator = nextItemIterator
          return true
        }
      }
      return true
    }
  }
}
internal class A sequence that
supports drop(n) and take(n) operations
internal interface DropTakeSequence<T> : Sequence<T> {
  fun drop(n: Int): Sequence<T>
  fun take(n: Int): Sequence<T>
}
internal class A sequence that skips
[startIndex] values from the underlying [sequence]
and stops returning values right before [endIndex], i.e. stops
at `endIndex - 1`
internal class SubSequence<T>(
private val sequence: Sequence<T>,
private
val startIndex: Int,
private val endIndex: Int) : Sequence<T>, DropTakeSequence<T> {
  init {
    require(startIndex >= 0) { "startIndex should be non-negative, but is $startIndex" }
    require(endIndex >=
0) { "endIndex should be non-negative, but is $endIndex" }
    require(endIndex >= startIndex) { "endIndex
should be not less than startIndex, but was $endIndex < $startIndex" }
  }
  private val count: Int get() =
endIndex - startIndex
  override fun drop(n: Int): Sequence<T> = if (n >= count) emptySequence() else
SubSequence(sequence, startIndex + n, endIndex)
  override fun take(n: Int): Sequence<T> = if (n >= count) this
else SubSequence(sequence, startIndex, startIndex + n)
  override fun iterator() = object : Iterator<T> {
    val iterator = sequence.iterator()
    var position = 0
    // Shouldn't be called from
constructor to avoid premature iteration
    private fun drop() {
      while (position < startIndex &&
iterator.hasNext()) {
        iterator.next()
        position++
      }
    }
    override
fun hasNext(): Boolean {
      drop()
      return (position < endIndex) && iterator.hasNext()
    }
    override fun next(): T {
      drop()
      if (position >= endIndex)
throw
NoSuchElementException()
      position++
      return iterator.next()
    }
  }
}

```

* A sequence that returns at most [count] values from the underlying [sequence], and stops returning values * as soon as that count is reached.

```

internal class TakeSequence<T>()
    private val sequence: Sequence<T>,
    private val count: Int : Sequence<T>, DropTakeSequence<T> {
        init {
            require(count >= 0) { "\"count must be non-negative, but was $count.\""}
        }
        override fun drop(n: Int): Sequence<T> = if (n >= count) emptySequence() else SubSequence(sequence, n, count)
        override fun take(n: Int): Sequence<T> = if (n >= count) this else TakeSequence(sequence, n)
        override fun iterator(): Iterator<T> = object : Iterator<T> {
            var left = count
            val iterator = sequence.iterator()
            override fun next(): T {
                if (left == 0) throw NoSuchElementException()
                left--
                return iterator.next()
            }
            override fun hasNext(): Boolean {
                return left > 0 && iterator.hasNext()
            }
        }
    }

```

* A sequence that returns values from the underlying [sequence] while the [predicate] function returns `true`, and stops returning values once the function returns `false` for the next element.

```

internal class TakeWhileSequence<T>()
    constructor(
        private val sequence: Sequence<T>,
        private val predicate: (T) -> Boolean : Sequence<T> {
        override fun iterator(): Iterator<T> = object : Iterator<T> {
            val iterator = sequence.iterator()
            var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue
            var nextItem: T? = null
            private fun calcNext() {
                if (iterator.hasNext()) {
                    val item = iterator.next()
                    if (predicate(item)) {
                        nextState = 1
                        nextItem = item
                        return
                    }
                }
                nextState = 0
            }
            override fun next(): T {
                if (nextState == -1) calcNext() // will change nextState
                if (nextState == 0) throw NoSuchElementException()
                @Suppress("UNCHECKED_CAST") val result = nextItem as T // Clean next to avoid keeping reference on yielded instance
                nextItem = null
                nextState = -1
                return result
            }
            override fun hasNext(): Boolean {
                if (nextState == -1) calcNext() // will change nextState
                return nextState == 1
            }
        }
    }

```

* A sequence that skips the specified number of values from the underlying [sequence] and returns all values after that.

```

internal class DropSequence<T>()
    private val sequence: Sequence<T>,
    private val count: Int : Sequence<T>,
    DropTakeSequence<T> {
        init {
            require(count >= 0) { "\"count must be non-negative, but was $count.\""}
        }
        override fun drop(n: Int): Sequence<T> = (count + n).let { n1 -> if (n1 < 0) DropSequence(this, n) else DropSequence(sequence, n1) }
        override fun take(n: Int): Sequence<T> = (count + n).let { n1 -> if (n1 < 0) TakeSequence(this, n) else SubSequence(sequence, count, n1) }
        override fun iterator(): Iterator<T> = object : Iterator<T> {
            val iterator = sequence.iterator()
            var left = count // Shouldn't be called from constructor to avoid premature iteration
            private fun drop() {
                while (left > 0 && iterator.hasNext()) {
                    iterator.next()
                    left--
                }
            }
            override fun next(): T {
                drop()
                return iterator.next()
            }
            override fun hasNext(): Boolean {
                drop()
                return iterator.hasNext()
            }
        }
    }

```

* A sequence that skips the values from the underlying [sequence] while the given [predicate] returns `true` and returns all values after that.

```

internal class DropWhileSequence<T>()
    constructor(
        private val sequence: Sequence<T>,
        private val predicate: (T) -> Boolean : Sequence<T> {
        override fun iterator(): Iterator<T> = object : Iterator<T> {
            val iterator = sequence.iterator()
            var dropState: Int = -1 // -1 for not dropping, 1 for nextItem, 0 for normal iteration
            var nextItem: T? = null
            private fun drop() {
                while (iterator.hasNext()) {
                    val item = iterator.next()
                    if (!predicate(item)) {
                        nextItem = item
                        dropState = 1
                        return
                    }
                }
                dropState = 0
            }
            override fun next(): T {
                if (dropState == -1) drop()
                if (dropState == 1) {
                    @Suppress("UNCHECKED_CAST") val result = nextItem as T
                    nextItem = null
                    dropState = 0
                    return result
                }
                return iterator.next()
            }
            override fun hasNext(): Boolean {
                if (dropState == -1) drop()
                return dropState == 1 || iterator.hasNext()
            }
        }
    }

```

```

internal class DistinctSequence<T, K>(
    private val source: Sequence<T>,
    private val keySelector: (T) -> K : Sequence<T> {
    override fun iterator(): Iterator<T> = DistinctIterator(source.iterator(), keySelector)
}
private class

```

```

DistinctIterator<T, K>(private val source: Iterator<T>, private val keySelector: (T) -> K) : AbstractIterator<T>()
{
    private val observed = HashSet<K>()
    override fun computeNext() {
        while (source.hasNext()) {
            val next = source.next()
            val key = keySelector(next)
            if (observed.add(key)) {
                setNext(next)
                return
            }
        }
        done()
    }
}

private class GeneratorSequence<T : Any>(private val getInitialValue: () -> T?, private val getNextValue: (T) -> T?) : Sequence<T> {
    override fun iterator(): Iterator<T> = object : Iterator<T> {
        var nextItem: T? = null
        var nextState: Int = -2 // -2 for initial unknown, -1 for next unknown, 0 for done, 1 for continue
        private fun calcNext() {
            nextItem = if (nextState == -2) getInitialValue() else getNextValue(nextItem!!)
            nextState = if (nextItem == null) 0 else 1
        }
        override fun next(): T {
            if (nextState < 0) {
                calcNext()
                if (nextState == 0) {
                    throw NoSuchElementException()
                }
                val result = nextItem as T
                // Do not clean nextItem (to avoid keeping reference on yielded instance)
                nextState = -1
                return result
            }
            override fun hasNext(): Boolean {
                if (nextState < 0) {
                    calcNext()
                    return nextState == 1
                }
            }
        }
    }
}

/** Returns a wrapper sequence that provides values of this sequence, but ensures it can be iterated only one time.
 * The operation is _intermediate_ and _stateless_.
 * [IllegalStateException] is thrown on iterating the returned sequence from the second time.
 */
public fun <T> Sequence<T>.constrainOnce(): Sequence<T> {
    // as? does not work in js
    //return this as? ConstrainedOnceSequence<T> ?: ConstrainedOnceSequence(this)
    return if (this is ConstrainedOnceSequence<T>) this else ConstrainedOnceSequence(this)
}

/** Returns a sequence which invokes the function to calculate the next value on each iteration until the function returns `null`.
 * The returned sequence is constrained to be iterated only once.
 * @see constrainOnce
 * @see kotlin.sequences.sequence
 * @sample samples.collections.Sequences.Building.generateSequence
 */
public fun <T : Any> generateSequence(nextFunction: () -> T?): Sequence<T> {
    return GeneratorSequence(nextFunction, { nextFunction() }).constrainOnce()
}

/** Returns a sequence defined by the starting value [seed] and the function [nextFunction],
 * which is invoked to calculate the next value based on the previous one on each iteration.
 * The sequence produces values until it encounters first `null` value.
 * If [seed] is `null`, an empty sequence is produced.
 * The sequence can be iterated multiple times, each time starting with [seed].
 * @see kotlin.sequences.sequence
 * @sample samples.collections.Sequences.Building.generateSequenceWithSeed
 */
@kotlin.internal.LowPriorityInOverloadResolution
public fun <T : Any> generateSequence(seed: T?, nextFunction: (T) -> T?): Sequence<T> =
    if (seed == null) EmptySequence
    else GeneratorSequence({ seed }, nextFunction)

/** Returns a sequence defined by the function [seedFunction], which is invoked to produce the starting value,
 * and the [nextFunction], which is invoked to calculate the next value based on the previous one on each iteration.
 * The sequence produces values until it encounters first `null` value.
 * If [seedFunction] returns `null`, an empty sequence is produced.
 * The sequence can be iterated multiple times.
 * @see kotlin.sequences.sequence
 * @sample samples.collections.Sequences.Building.generateSequenceWithLazySeed
 */
public fun <T : Any> generateSequence(seedFunction: () -> T?, nextFunction: (T) -> T?): Sequence<T> =
    GeneratorSequence(seedFunction, nextFunction)
}

/** Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("PreconditionsKt")
package kotlin
import kotlin.contracts.contract
/** Throws an [IllegalArgumentException] if the [value] is false.
 * @sample samples.misc.Preconditions.failRequireWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun require(value: Boolean): Unit {
    contract {
        returns() implies value
    }
    require(value) { "Failed requirement." }
}

/** Throws an [IllegalArgumentException] with the result of calling [lazyMessage] if the [value] is false.
 * @sample samples.misc.Preconditions.failRequireWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun

```

```

require(value: Boolean, lazyMessage: () -> Any): Unit {
    contract {
        returns() implies value
    }
    if (!value) {
        val message = lazyMessage()
        throw IllegalArgumentException(message.toString())
    }
}

Throws an [IllegalArgumentException] if the [value] is null. Otherwise returns the not null value.

@kotlin.internal.InlineOnly
public inline fun <T : Any> requireNotNull(value: T?): T {
    contract {
        returns() implies (value != null)
    }
    return requireNotNull(value) { "Required value was null." }
}

Throws an [IllegalArgumentException] with the result of calling [lazyMessage] if the [value] is null. Otherwise returns the not null value.

@sample
samples.misc.Preconditions.failRequireWithLazyMessage

@kotlin.internal.InlineOnly
public inline fun <T : Any> requireNotNull(value: T?, lazyMessage: () -> Any): T {
    contract {
        returns() implies (value != null)
    }
    if (value == null) {
        val message = lazyMessage()
        throw IllegalArgumentException(message.toString())
    } else {
        return value
    }
}

Throws an [IllegalStateException] if the [value] is false.

@sample
samples.misc.Preconditions.failCheckWithLazyMessage

@kotlin.internal.InlineOnly
public inline fun check(value: Boolean): Unit {
    contract {
        returns() implies value
    }
    check(value) { "Check failed." }
}

Throws an [IllegalStateException] with the result of calling [lazyMessage] if the [value] is false.

@sample
samples.misc.Preconditions.failCheckWithLazyMessage

@kotlin.internal.InlineOnly
public inline fun check(value: Boolean, lazyMessage: () -> Any): Unit {
    contract {
        returns() implies value
    }
    if (!value) {
        val message = lazyMessage()
        throw IllegalStateException(message.toString())
    }
}

Throws an [IllegalStateException] if the [value] is null. Otherwise returns the not null value.

@sample
samples.misc.Preconditions.failCheckWithLazyMessage

@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?): T {
    contract {
        returns() implies (value != null)
    }
    return checkNotNull(value) { "Required value was null." }
}

Throws an [IllegalStateException] with the result of calling [lazyMessage] if the [value] is null. Otherwise returns the not null value.

@sample
samples.misc.Preconditions.failCheckWithLazyMessage

@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, lazyMessage: () -> Any): T {
    contract {
        returns() implies (value != null)
    }
    if (value == null) {
        val message = lazyMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

Throws an [IllegalStateException] with the given [message].

@sample
samples.misc.Preconditions.failWithError

@kotlin.internal.InlineOnly
public inline fun error(message: Any): Nothing = throw IllegalStateException(message.toString())

/*
 * Copyright 2010-2019 JetBrains s.r.o.
 * and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license
 * that can be found in the license/LICENSE.txt file.
 */
package kotlin.collections

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.js.*
import primitiveArrayConcat
import withType
import kotlin.ranges.contains
import kotlin.ranges.reversed

Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample
samples.collections.Collections.Elements.elementAt

public actual fun <T> Array<out T>.elementAt(index: Int): T {
    return elementAtOrElse(index) { throw
IndexOutOfBoundsException("index: $index, size: $size") }
}

Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample
samples.collections.Collections.Elements.elementAt

public actual fun ByteArray.elementAt(index: Int): Byte {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample
samples.collections.Collections.Elements.elementAt

public actual fun ShortArray.elementAt(index: Int): Short {
    return elementAtOrElse(index) { throw
IndexOutOfBoundsException("index: $index, size: $size") }
}

Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample

```

```

samples.collections.Collections.Elements.elementAt\n *^npublic actual fun IntArray.elementAt(index: Int): Int {\n
return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size}") }\n}\n\n/**\n
* Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds
of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n *^npublic actual fun
LongArray.elementAt(index: Int): Long {\n  return elementAtOrElse(index) { throw
IndexOutOfBoundsException("index: $index, size: $size}") }\n}\n\n/**\n * Returns an element at the given
[index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAt\n *^npublic actual fun FloatArray.elementAt(index: Int): Float
{\n  return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size}")
}\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is
out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n *^npublic
actual fun DoubleArray.elementAt(index: Int): Double {\n  return elementAtOrElse(index) { throw
IndexOutOfBoundsException("index: $index, size: $size}") }\n}\n\n/**\n * Returns an element at the given
[index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAt\n *^npublic actual fun BooleanArray.elementAt(index: Int):
Boolean {\n  return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size")
}\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is
out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n *^npublic
actual fun CharArray.elementAt(index: Int): Char {\n  return elementAtOrElse(index) { throw
IndexOutOfBoundsException("index: $index, size: $size") }\n}\n\n/**\n * Returns a [List] that wraps the original
array.\n *^npublic actual fun <T> Array<out T>.asList(): List<T> {\n  return
ArrayList<T>(this.unsafeCast<Array<Any?>>())\n}\n\n/**\n * Returns a [List] that wraps the original array.\n
*^n@kotlin.internal.InlineOnly\npublic actual inline fun ByteArray.asList(): List<Byte> {\n  return
this.unsafeCast<Array<Byte>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n
*^n@kotlin.internal.InlineOnly\npublic actual inline fun ShortArray.asList(): List<Short> {\n  return
this.unsafeCast<Array<Short>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n
*^n@kotlin.internal.InlineOnly\npublic actual inline fun IntArray.asList(): List<Int> {\n  return
this.unsafeCast<Array<Int>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n
*^n@kotlin.internal.InlineOnly\npublic actual inline fun LongArray.asList(): List<Long> {\n  return
this.unsafeCast<Array<Long>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n
*^n@kotlin.internal.InlineOnly\npublic actual inline fun FloatArray.asList(): List<Float> {\n  return
this.unsafeCast<Array<Float>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n
*^n@kotlin.internal.InlineOnly\npublic actual inline fun DoubleArray.asList(): List<Double> {\n  return
this.unsafeCast<Array<Double>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n
*^n@kotlin.internal.InlineOnly\npublic actual inline fun BooleanArray.asList(): List<Boolean> {\n  return
this.unsafeCast<Array<Boolean>>().asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n
*^npublic actual fun CharArray.asList(): List<Char> {\n  return object : ArrayList<Char>(), RandomAccess {\n
  override val size: Int get() = this@asList.size\n  override fun isEmpty(): Boolean = this@asList.isEmpty()\n
  override fun contains(element: Char): Boolean = this@asList.contains(element)\n  override fun get(index: Int):
Char {\n    ArrayList.checkElementIndex(index, size)\n    return this@asList[index]\n  }\n  override fun
indexOf(element: Char): Int {\n    if ((element as Any?) !is Char) return -1\n    return
this@asList.indexOf(element)\n  }\n  override fun lastIndexOf(element: Char): Int {\n    if ((element as
Any?) !is Char) return -1\n    return this@asList.lastIndexOf(element)\n  }\n}\n}\n\n/**\n * Returns
`true` if the two specified arrays are *deeply* equal to one another,\n * i.e. contain the same number of the same
elements in the same order.\n * \n * If two corresponding elements are nested arrays, they are also compared
deeply.\n * \n * If any of arrays contains itself on any nesting level the behavior is undefined.\n * \n * The elements of
other types are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it
means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n

```

```

*\n@SinceKotlin("1.1")\n@library("arrayDeepEquals")\npublic actual infix fun <T> Array<out
T>.contentDeepEquals(other: Array<out T>): Boolean {\n  definedExternally\n}\n\n/**\n * Returns a hash code
based on the contents of this array as if it is [List].\n * Nested arrays are treated as lists too.\n * \n * If any of arrays
contains itself on any nesting level the behavior is undefined.\n
*\n@SinceKotlin("1.1")\n@library("arrayDeepHashCode")\npublic actual fun <T> Array<out
T>.contentDeepHashCode(): Int {\n  definedExternally\n}\n\n/**\n * Returns a string representation of the
contents of this array as if it is a [List].\n * Nested arrays are treated as lists too.\n * \n * If any of arrays contains
itself on any nesting level that reference\n * is rendered as `"[...]"` to prevent recursion.\n * \n * @sample
samples.collections.Arrays.ContentOperations.contentDeepToString\n
*\n@SinceKotlin("1.1")\n@library("arrayDeepToString")\npublic actual fun <T> Array<out
T>.contentDeepToString(): String {\n  definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays
are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n *
\n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers
it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n
*\n@SinceKotlin("1.1")\n@library("arrayEquals")\npublic actual infix fun <T> Array<out
T>.contentEquals(other: Array<out T>): Boolean {\n  definedExternally\n}\n\n/**\n * Returns `true` if the two
specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the
same order.\n * \n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating
point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n
*\n@SinceKotlin("1.1")\n@library("arrayEquals")\npublic actual infix fun ByteArray.contentEquals(other:
ByteArray): Boolean {\n  definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are
*structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n
* The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it
means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n
*\n@SinceKotlin("1.1")\n@library("arrayEquals")\npublic actual infix fun ShortArray.contentEquals(other:
ShortArray): Boolean {\n  definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are
*structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n
* The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it
means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n
*\n@SinceKotlin("1.1")\n@library("arrayEquals")\npublic actual infix fun IntArray.contentEquals(other:
IntArray): Boolean {\n  definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are *structurally*
equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * The elements
are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that
`NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n
*\n@SinceKotlin("1.1")\n@library("arrayEquals")\npublic actual infix fun LongArray.contentEquals(other:
LongArray): Boolean {\n  definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are
*structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n
* The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it
means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n
*\n@SinceKotlin("1.1")\n@library("arrayEquals")\npublic actual infix fun FloatArray.contentEquals(other:
FloatArray): Boolean {\n  definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are
*structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n
* The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it
means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n
*\n@SinceKotlin("1.1")\n@library("arrayEquals")\npublic actual infix fun DoubleArray.contentEquals(other:
DoubleArray): Boolean {\n  definedExternally\n}\n\n/**\n * Returns `true` if the two specified arrays are
*structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n
* The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it
means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n

```

means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```
*\n@SinceKotlin("1.1")\n@library("arrayEquals")\npublic actual infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean {\n    definedExternally\n}\n/**\n * Returns `true` if the two specified arrays are\n * structurally equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
```

```
*\n@SinceKotlin("1.1")\n@library("arrayEquals")\npublic actual infix fun CharArray.contentEquals(other: CharArray): Boolean {\n    definedExternally\n}\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n *\n@SinceKotlin("1.1")\n@library("arrayHashCode")\npublic actual fun <T> Array<out T>.contentHashCode(): Int {\n    definedExternally\n}\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n *\n@SinceKotlin("1.1")\n@library("arrayHashCode")\npublic actual fun ByteArray.contentHashCode(): Int {\n    definedExternally\n}\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n *\n@SinceKotlin("1.1")\n@library("arrayHashCode")\npublic actual fun ShortArray.contentHashCode(): Int {\n    definedExternally\n}\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n *\n@SinceKotlin("1.1")\n@library("arrayHashCode")\npublic actual fun IntArray.contentHashCode(): Int {\n    definedExternally\n}\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n *\n@SinceKotlin("1.1")\n@library("arrayHashCode")\npublic actual fun LongArray.contentHashCode(): Int {\n    definedExternally\n}\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n *\n@SinceKotlin("1.1")\n@library("arrayHashCode")\npublic actual fun FloatArray.contentHashCode(): Int {\n    definedExternally\n}\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n *\n@SinceKotlin("1.1")\n@library("arrayHashCode")\npublic actual fun DoubleArray.contentHashCode(): Int {\n    definedExternally\n}\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n *\n@SinceKotlin("1.1")\n@library("arrayHashCode")\npublic actual fun BooleanArray.contentHashCode(): Int {\n    definedExternally\n}\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n *\n@SinceKotlin("1.1")\n@library("arrayHashCode")\npublic actual fun CharArray.contentHashCode(): Int {\n    definedExternally\n}\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n *\n * @sample\n samples.collections.Arrays.ContentOperations.contentToString\n
```

```
*\n@SinceKotlin("1.1")\n@library("arrayToString")\npublic actual fun <T> Array<out T>.contentToString(): String {\n    definedExternally\n}\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n *\n * @sample\n samples.collections.Arrays.ContentOperations.contentToString\n
```

```
*\n@SinceKotlin("1.1")\n@library("arrayToString")\npublic actual fun ByteArray.contentToString(): String {\n    definedExternally\n}\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n *\n * @sample\n samples.collections.Arrays.ContentOperations.contentToString\n
```

```
*\n@SinceKotlin("1.1")\n@library("arrayToString")\npublic actual fun ShortArray.contentToString(): String {\n    definedExternally\n}\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n *\n * @sample\n samples.collections.Arrays.ContentOperations.contentToString\n
```

```
*\n@SinceKotlin("1.1")\n@library("arrayToString")\npublic actual fun IntArray.contentToString(): String {\n    definedExternally\n}\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n *\n * @sample\n samples.collections.Arrays.ContentOperations.contentToString\n
```

```
*\n@SinceKotlin("1.1")\n@library("arrayToString")\npublic actual fun LongArray.contentToString(): String {\n    definedExternally\n}\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n *\n * @sample\n samples.collections.Arrays.ContentOperations.contentToString\n
```

```
*\n@SinceKotlin("1.1")\n@library("arrayToString")\npublic actual fun FloatArray.contentToString(): String {\n    definedExternally\n}\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n *\n * @sample\n samples.collections.Arrays.ContentOperations.contentToString\n
```

```
*\n@SinceKotlin("1.1")\n@library("arrayToString")\npublic actual fun DoubleArray.contentToString(): String {\n    definedExternally\n}\n/**\n * Returns a string representation of the contents of the specified array as if it is
```



```

[List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n
*\n@SinceKotlin("1.1")\n@library("arrayToString")\npublic actual fun BooleanArray.contentToString(): String
{\n  definedExternally\n}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is
[List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n
*\n@SinceKotlin("1.1")\n@library("arrayToString")\npublic actual fun CharArray.contentToString(): String {\n
  definedExternally\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that
array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it
overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset
the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of
the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this
array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex]
or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws
IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified
[destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the
[destination] array.\n
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT
_ARGUMENTS")\npublic actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset:
Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {\n  arrayCopy(this, destination, destinationOffset,
startIndex, endIndex)\n  return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination]
array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the
subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n *
@param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the
beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the
subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or
[IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex
> endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array
starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices
range.\n * \n * @return the [destination] array.\n
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT
_ARGUMENTS")\npublic actual inline fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0,
startIndex: Int = 0, endIndex: Int = size): ByteArray {\n  arrayCopy(this.unsafeCast<Array<Byte>>(),
destination.unsafeCast<Array<Byte>>(), destinationOffset, startIndex, endIndex)\n  return destination\n}\n\n/**\n
 * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass
the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n
 * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array
to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws
IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this
array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange
doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out
of the [destination] array indices range.\n * \n * @return the [destination] array.\n
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT
_ARGUMENTS")\npublic actual inline fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int =
0, startIndex: Int = 0, endIndex: Int = size): ShortArray {\n  arrayCopy(this.unsafeCast<Array<Short>>(),
destination.unsafeCast<Array<Short>>(), destinationOffset, startIndex, endIndex)\n  return destination\n}\n\n/**\n
 * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass
the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n
 * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array
to

```

copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n

```
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray {\n    arrayCopy(this.unsafeCast<Array<Int>>(), destination.unsafeCast<Array<Int>>(), destinationOffset, startIndex, endIndex)\n    return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n
```

```
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): LongArray {\n    arrayCopy(this.unsafeCast<Array<Long>>(), destination.unsafeCast<Array<Long>>(), destinationOffset, startIndex, endIndex)\n    return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n
```

```
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): FloatArray {\n    arrayCopy(this.unsafeCast<Array<Float>>(), destination.unsafeCast<Array<Float>>(), destinationOffset, startIndex, endIndex)\n    return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n
```

```
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): DoubleArray {\n    arrayCopy(this.unsafeCast<Array<Double>>(),
```

```

destination.unsafeCast<Array<Double>>(), destinationOffset, startIndex, endIndex)\n    return
destination\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n *
It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the
[destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy,
0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n *
@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the
subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is
out of the [destination] array indices range.\n * \n * @return the [destination] array.\n
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT
_ARGUMENTS")\npublic actual inline fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset:
Int = 0, startIndex: Int = 0, endIndex: Int = size): BooleanArray {\n
arrayCopy(this.unsafeCast<Array<Boolean>>(), destination.unsafeCast<Array<Boolean>>(), destinationOffset,
startIndex, endIndex)\n    return destination\n}\n\n/**\n * Copies this array or its subrange into the [destination]
array and returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the
subrange so that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n *
@param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the
beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the
subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or
[IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex
> endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array
starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n *
\n * @return the [destination] array.\n
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT
_ARGUMENTS")\npublic actual inline fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0,
startIndex: Int = 0, endIndex: Int = size): CharArray {\n    arrayCopy(this.unsafeCast<Array<Char>>(),
destination.unsafeCast<Array<Char>>(), destinationOffset, startIndex, endIndex)\n    return destination\n}\n\n/**\n *
Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n *\n@Suppress("ACTUAL_WITHOUT_EXPECT",
"NOTHING_TO_INLINE")\npublic actual inline fun <T> Array<out T>.copyOf(): Array<T> {\n    return
this.asDynamic().slice()\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n *\n@Suppress("NOTHING_TO_INLINE")\npublic
actual inline fun ByteArray.copyOf(): ByteArray {\n    return this.asDynamic().slice()\n}\n\n/**\n * Returns new
array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n *\n@Suppress("NOTHING_TO_INLINE")\npublic
actual inline fun ShortArray.copyOf(): ShortArray {\n    return this.asDynamic().slice()\n}\n\n/**\n * Returns new
array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n *\n@Suppress("NOTHING_TO_INLINE")\npublic
actual inline fun IntArray.copyOf(): IntArray {\n    return this.asDynamic().slice()\n}\n\n/**\n * Returns new array
which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOf\n
*\n@Suppress("NOTHING_TO_INLINE")\npublic
actual inline fun LongArray.copyOf(): LongArray {\n    return withType("LongArray",
this.asDynamic().slice()\n}\n\n/**\n * Returns new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n *\n@Suppress("NOTHING_TO_INLINE")\npublic
actual inline fun FloatArray.copyOf(): FloatArray {\n    return this.asDynamic().slice()\n}\n\n/**\n * Returns new
array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n *\n@Suppress("NOTHING_TO_INLINE")\npublic
actual inline fun DoubleArray.copyOf(): DoubleArray {\n    return this.asDynamic().slice()\n}\n\n/**\n * Returns

```

```

new array which is a copy of the original array.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.copyOfOf\n * \n\npublic actual fun BooleanArray.copyOfOf():
BooleanArray {\n * return withType("\BooleanArray", this.asDynamic().slice())\n}\n\n/**\n * Returns new array
which is a copy of the original array.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.copyOfOf\n
*\n\npublic actual fun CharArray.copyOfOf(): CharArray {\n * return withType("\CharArray",
this.asDynamic().slice())\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given
[newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n * \n * - If [newSize]
is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater
than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n *\n\npublic actual fun
ByteArray.copyOfOf(newSize: Int): ByteArray {\n * require(newSize >= 0) { "\Invalid new array size: $newSize.\n"
}\n * return fillFrom(this, ByteArray(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original
array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if
necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the
[newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are
filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n
*\n\npublic actual fun ShortArray.copyOfOf(newSize: Int): ShortArray {\n * require(newSize >= 0) { "\Invalid new
array size: $newSize.\n" }\n * return fillFrom(this, ShortArray(newSize))\n}\n\n/**\n * Returns new array which is a
copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with
zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated
to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array
are filled with zero values.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n *\n\npublic actual fun
IntArray.copyOfOf(newSize: Int): IntArray {\n * require(newSize >= 0) { "\Invalid new array size: $newSize.\n" }\n
return fillFrom(this, IntArray(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array,
resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if necessary.\n
*\n * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n * - If
[newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero
values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n *\n\npublic actual
fun LongArray.copyOfOf(newSize: Int): LongArray {\n * require(newSize >= 0) { "\Invalid new array size:
$newSize.\n" }\n * return withType("\LongArray", arrayCopyResize(this, newSize, 0L))\n}\n\n/**\n * Returns new
array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded
at the end with zero values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy
array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements
in the copy array are filled with zero values.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n *\n\npublic actual fun
FloatArray.copyOfOf(newSize: Int): FloatArray {\n * require(newSize >= 0) { "\Invalid new array size: $newSize.\n"
}\n * return fillFrom(this, FloatArray(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original
array, resized to the given [newSize].\n * The copy is either truncated or padded at the end with zero values if
necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is truncated to the
[newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are
filled with zero values.\n * \n * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n
*\n\npublic actual fun DoubleArray.copyOfOf(newSize: Int): DoubleArray {\n * require(newSize >= 0) { "\Invalid new
array size: $newSize.\n" }\n * return fillFrom(this, DoubleArray(newSize))\n}\n\n/**\n * Returns new array which is
a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end
with `false` values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is
truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the
copy array are filled with `false` values.\n * \n * @sample

```

```

samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n *^npublic actual fun
BooleanArray.copyOf(newSize: Int): BooleanArray {\n  require(newSize >= 0) { \"Invalid new array size:
$newSize.\" }\n  return withType(\"BooleanArray\", arrayCopyResize(this, newSize, false))\n}\n\n/**\n * Returns
new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or
padded at the end with null char (``\u0000``) values if necessary.\n * \n * - If [newSize] is less than the size of the
original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original
array, the extra elements in the copy array are filled with null char (``\u0000``) values.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n *^npublic actual fun
CharArray.copyOf(newSize: Int): CharArray {\n  require(newSize >= 0) { \"Invalid new array size: $newSize.\"
}\n  return withType(\"CharArray\", fillFrom(this, CharArray(newSize)))\n}\n\n/**\n * Returns new array which is
a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at the end
with `null` values if necessary.\n * \n * - If [newSize] is less than the size of the original array, the copy array is
truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the
copy array are filled with `null` values.\n * \n * @sample
samples.collections.Arrays.CopyOfOperations.resizingCopyOf\n
*^n@Suppress(\"ACTUAL_WITHOUT_EXPECT\")\npublic actual fun <T> Array<out T>.copyOf(newSize: Int):
Array<T?> {\n  require(newSize >= 0) { \"Invalid new array size: $newSize.\" }\n  return arrayCopyResize(this,
newSize, null)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n
* @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the
range (exclusive), must be in `fromIndex..array.size`\n
*^n@Suppress(\"ACTUAL_WITHOUT_EXPECT\")\npublic actual fun <T> Array<out
T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T> {\n  AbstractList.checkRangeIndexes(fromIndex,
toIndex, size)\n  return this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a
copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must
be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n
*^npublic actual fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray {\n
AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n  return this.asDynamic().slice(fromIndex,
toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n *
@param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the
range (exclusive), must be in `fromIndex..array.size`\n *^npublic actual fun ShortArray.copyOfRange(fromIndex:
Int, toIndex: Int): ShortArray {\n  AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n  return
this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range
of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n *
@param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n *^npublic actual fun
IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray {\n  AbstractList.checkRangeIndexes(fromIndex,
toIndex, size)\n  return this.asDynamic().slice(fromIndex, toIndex)\n}\n\n/**\n * Returns a new array which is a
copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must
be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n
*^npublic actual fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray {\n
AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n  return withType(\"LongArray\",
this.asDynamic().slice(fromIndex, toIndex))\n}\n\n/**\n * Returns a new array which is a copy of the specified
range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n
* @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n *^npublic actual fun
FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray {\n
AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n  return this.asDynamic().slice(fromIndex,
toIndex)\n}\n\n/**\n * Returns a new array which is a copy of the specified range of the original array.\n * \n *
@param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the
range (exclusive), must be in `fromIndex..array.size`\n *^npublic actual fun DoubleArray.copyOfRange(fromIndex:

```

```

Int, toIndex: Int): DoubleArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    return
    this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns a new array which is a copy of the specified range
 * of the original array.
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */
public actual fun
BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    return withType("BooleanArray"),
    this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns a new array which is a copy of the specified
 * range of the original array.
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */
public actual fun
CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    return withType("CharArray"),
    this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified
 * [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if
 * [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws
 * IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun <T> Array<T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex,
    toIndex);
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param
 * fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size
 * of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex]
 * is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than
 * [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex,
    toIndex);
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param
 * fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size
 * of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex]
 * is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than
 * [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex,
    toIndex);
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param
 * fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size
 * of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex]
 * is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than
 * [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex,
    toIndex);
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param
 * fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size
 * of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex]
 * is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than
 * [toIndex].
 */

```

```

*\n@SinceKotlin("1.3")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic
actual fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit {\n
AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n  this.asDynamic().fill(element, fromIndex,
toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param
fromIndex the start of the range (inclusive), 0 by default.\n * @param toIndex the end of the range (exclusive), size
of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex]
is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than
[toIndex].\n
*\n@SinceKotlin("1.3")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic
actual fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit {\n
AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n  this.asDynamic().fill(element, fromIndex,
toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param
fromIndex the start of the range (inclusive), 0 by default.\n * @param toIndex the end of the range (exclusive), size
of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex]
is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than
[toIndex].\n
*\n@SinceKotlin("1.3")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic
actual fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit {\n
AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n  this.asDynamic().fill(element, fromIndex,
toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param
fromIndex the start of the range (inclusive), 0 by default.\n * @param toIndex the end of the range (exclusive), size
of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex]
is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than
[toIndex].\n
*\n@SinceKotlin("1.3")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic
actual fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit {\n
AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n  this.asDynamic().fill(element, fromIndex,
toIndex);\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param
fromIndex the start of the range (inclusive), 0 by default.\n * @param toIndex the end of the range (exclusive), size
of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex]
is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than
[toIndex].\n
*\n@SinceKotlin("1.3")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic
actual fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit {\n
AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n  this.asDynamic().fill(element, fromIndex,
toIndex);\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n
*\n@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")\npublic actual inline operator
fun <T> Array<out T>.plus(element: T): Array<T> {\n  return
this.asDynamic().concat(arrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original
array and then the given [element].\n *\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline operator
fun ByteArray.plus(element: Byte): ByteArray {\n  return plus(byteArrayOf(element))\n}\n\n/**\n * Returns an
array containing all elements of the original array and then the given [element].\n
*\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun ShortArray.plus(element: Short):
ShortArray {\n  return plus(shortArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the
original array and then the given [element].\n *\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline
operator fun IntArray.plus(element: Int): IntArray {\n  return plus(intArrayOf(element))\n}\n\n/**\n * Returns an
array containing all elements of the original array and then the given [element].\n
*\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun LongArray.plus(element: Long):

```

LongArray { \n return plus(longArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun FloatArray.plus(element: Float): FloatArray { \n return plus(floatArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun DoubleArray.plus(element: Double): DoubleArray { \n return plus(doubleArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun BooleanArray.plus(element: Boolean): BooleanArray { \n return plus(booleanArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given [element].\n */\n\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun CharArray.plus(element: Char): CharArray { \n return plus(charArrayOf(element))\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n\n@Suppress("ACTUAL_WITHOUT_EXPECT")\npublic actual operator fun <T> Array<out T>.plus(elements: Collection<T>): Array<T> { \n return arrayPlusCollection(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n\n@public actual operator fun ByteArray.plus(elements: Collection<Byte>): ByteArray { \n return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n\n@public actual operator fun ShortArray.plus(elements: Collection<Short>): ShortArray { \n return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n\n@public actual operator fun IntArray.plus(elements: Collection<Int>): IntArray { \n return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n\n@public actual operator fun LongArray.plus(elements: Collection<Long>): LongArray { \n return arrayPlusCollection(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n\n@public actual operator fun FloatArray.plus(elements: Collection<Float>): FloatArray { \n return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n\n@public actual operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray { \n return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n\n@public actual operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray { \n return arrayPlusCollection(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\n\n@public actual operator fun CharArray.plus(elements: Collection<Char>): CharArray { \n return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n\n@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")\npublic actual inline operator fun <T> Array<out T>.plus(elements: Array<out T>): Array<T> { \n return this.asDynamic().concat(elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun ByteArray.plus(elements: ByteArray): ByteArray { \n return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun ShortArray.plus(elements: ShortArray): ShortArray { \n return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n */\n\n@Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun IntArray.plus(elements: IntArray):


```

IntArray { \n  return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing all elements of
the original array and then all elements of the given [elements] array.\n
*\n@\Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun LongArray.plus(elements:
LongArray): LongArray { \n  return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array
containing all elements of the original array and then all elements of the given [elements] array.\n
*\n@\Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun FloatArray.plus(elements:
FloatArray): FloatArray { \n  return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing
all elements of the original array and then all elements of the given [elements] array.\n
*\n@\Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun DoubleArray.plus(elements:
DoubleArray): DoubleArray { \n  return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array
containing all elements of the original array and then all elements of the given [elements] array.\n
*\n@\Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun BooleanArray.plus(elements:
BooleanArray): BooleanArray { \n  return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array
containing all elements of the original array and then all elements of the given [elements] array.\n
*\n@\Suppress("NOTHING_TO_INLINE")\npublic actual inline operator fun CharArray.plus(elements:
CharArray): CharArray { \n  return primitiveArrayConcat(this, elements)\n}\n\n/**\n * Returns an array containing
all elements of the original array and then the given [element].\n
*\n@\Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")\npublic actual inline fun <T>
Array<out T>.plusElement(element: T): Array<T> { \n  return
this.asDynamic().concat(arrayOf(element))\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n *\n@\library("primitiveArraySort")\npublic actual fun
IntArray.sort(): Unit { \n  definedExternally\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n *\npublic actual fun LongArray.sort(): Unit { \n  if (size > 1) sort {
a: Long, b: Long -> a.compareTo(b) }\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n *\n@\library("primitiveArraySort")\npublic actual fun
ByteArray.sort(): Unit { \n  definedExternally\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n *\n@\library("primitiveArraySort")\npublic actual fun
ShortArray.sort(): Unit { \n  definedExternally\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n *\n@\library("primitiveArraySort")\npublic actual fun
DoubleArray.sort(): Unit { \n  definedExternally\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n *\n@\library("primitiveArraySort")\npublic actual fun
FloatArray.sort(): Unit { \n  definedExternally\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n *\n@\library("primitiveArraySort")\npublic actual fun
CharArray.sort(): Unit { \n  definedExternally\n}\n\n/**\n * Sorts the array in-place according to the natural order
of its elements.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other
after sorting.\n * \n * @sample samples.collections.Arrays.Sorting.sortArrayOfComparable\n *\npublic actual fun
<T : Comparable<T>> Array<out T>.sort(): Unit { \n  if (size > 1) sortArray(this)\n}\n\n/**\n * Sorts the array in-
place according to the order specified by the given [comparison] function.\n * \n * The sort is _stable_. It means that
equal elements preserve their order relative to each other after sorting.\n *\npublic fun <T> Array<out
T>.sort(comparison: (a: T, b: T) -> Int): Unit { \n  if (size > 1) sortArrayWith(this, comparison)\n}\n\n/**\n * Sorts
the array in-place according to the order specified by the given [comparison] function.\n
*\n@\kotlin.internal.InlineOnly\npublic inline fun ByteArray.sort(noinline comparison: (a: Byte, b: Byte) -> Int):
Unit { \n  asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according to the order specified by
the given [comparison] function.\n *\n@\kotlin.internal.InlineOnly\npublic inline fun ShortArray.sort(noinline
comparison: (a: Short, b: Short) -> Int): Unit { \n  asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-
place according to the order specified by the given [comparison] function.\n *\n@\kotlin.internal.InlineOnly\npublic
inline fun IntArray.sort(noinline comparison: (a: Int, b: Int) -> Int): Unit { \n
asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given

```

```

[comparison] function.\n *^@kotlin.internal.InlineOnly\npublic inline fun LongArray.sort(noinline comparison:
(a: Long, b: Long) -> Int): Unit {\n  asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according
to the order specified by the given [comparison] function.\n *^@kotlin.internal.InlineOnly\npublic inline fun
FloatArray.sort(noinline comparison: (a: Float, b: Float) -> Int): Unit {\n
asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place according to the order specified by the given
[comparison] function.\n *^@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.sort(noinline comparison:
(a: Double, b: Double) -> Int): Unit {\n  asDynamic().sort(comparison)\n}\n\n/**\n * Sorts the array in-place
according to the order specified by the given [comparison] function.\n *^@kotlin.internal.InlineOnly\npublic
inline fun CharArray.sort(noinline comparison: (a: Char, b: Char) -> Int): Unit {\n
asDynamic().sort(comparison)\n}\n\n/**\n * The sort is _stable_. It means that equal elements preserve their order relative to each other
after sorting.\n *^npublic actual fun <T> Array<out T>.sortWith(comparator: Comparator<in T>): Unit {\n  if
(size > 1) sortArrayWith(this, comparator)\n}\n\n/**\n * Returns a *typed* object array containing all of the
elements of this primitive array.\n *^npublic actual fun ByteArray.toTypedArray(): Array<Byte> {\n  return
js("\\[\\]").slice.call(this)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive
array.\n *^npublic actual fun ShortArray.toTypedArray(): Array<Short> {\n  return
js("\\[\\]").slice.call(this)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive
array.\n *^npublic actual fun IntArray.toTypedArray(): Array<Int> {\n  return js("\\[\\]").slice.call(this)\n}\n\n/**\n * Returns a *typed* object array containing all of the elements of this primitive array.\n *^npublic actual fun
LongArray.toTypedArray(): Array<Long> {\n  return js("\\[\\]").slice.call(this)\n}\n\n/**\n * Returns a *typed*
object array containing all of the elements of this primitive array.\n *^npublic actual fun
FloatArray.toTypedArray(): Array<Float> {\n  return js("\\[\\]").slice.call(this)\n}\n\n/**\n * Returns a *typed*
object array containing all of the elements of this primitive array.\n *^npublic actual fun
DoubleArray.toTypedArray(): Array<Double> {\n  return js("\\[\\]").slice.call(this)\n}\n\n/**\n * Returns a *typed*
object array containing all of the elements of this primitive array.\n *^npublic actual fun
BooleanArray.toTypedArray(): Array<Boolean> {\n  return js("\\[\\]").slice.call(this)\n}\n\n/**\n * Returns a
*typed* object array containing all of the elements of this primitive array.\n *^npublic actual fun
CharArray.toTypedArray(): Array<Char> {\n  return Array(size) { index -> this[index] }\n}\n\n"/**\n *
Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code
is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n *^@file:kotlin.jvm.JvmName("ComparisonsKt")\n *^@file:kotlin.jvm.JvmMultifileClass\n *^npackage
kotlin.comparisons\n *^n\n/**\n * Compares two values using the specified functions [selectors] to calculate the
result of the comparison.\n * The functions are called sequentially, receive the given values [a] and [b] and return
[Comparable]\n * objects. As soon as the [Comparable] instances returned by a function for [a] and [b] values do
not\n * compare as equal, the result of that comparison is returned.\n *^r\n * @sample
samples.comparisons.Comparisons.compareValuesByWithSelectors\n *^r\npublic fun <T> compareValuesBy(a:
T, b: T, vararg selectors: (T) -> Comparable<*>?): Int {\n  require(selectors.size > 0)\n  return
compareValuesByImpl(a, b, selectors)\n}\n\nprivate fun <T> compareValuesByImpl(a: T, b: T, selectors:
Array<out (T) -> Comparable<*>?): Int {\n  for (fn in selectors) {\n    val v1 = fn(a)\n    val v2 =
fn(b)\n    val diff = compareValues(v1, v2)\n    if (diff != 0) return diff\n  }\n  return
0\n}\n\n/**\n * Compares two values using the specified [selector] function to calculate the result of the
comparison.\n * The function is applied to the given values [a] and [b] and return [Comparable] objects.\n * The
result of comparison of these [Comparable] instances is returned.\n *^r\n * @sample
samples.comparisons.Comparisons.compareValuesByWithSingleSelector\n *^r\n *^@kotlin.internal.InlineOnly\npublic inline fun <T> compareValuesBy(a: T, b: T, selector: (T) ->
Comparable<*>?): Int {\n  return compareValues(selector(a), selector(b))\n}\n\n/**\n * Compares two
values using the specified [selector] function to calculate the result of the comparison.\n * The function is applied
to the given values [a] and [b] and return objects of type K which are then being\n * compared with the given

```



```

(previousCompare != 0) previousCompare else compareValuesBy(a, b, comparator, selector)\r\n    }\r\n\r\n/**\r\n *
Creates a descending comparator using the primary comparator and\r\n * the function to transform value to a
[Comparable] instance for comparison.\r\n * \r\n * @sample
samples.comparisons.Comparisons.thenByDescending\r\n * \r\n * @kotlin.internal.InlineOnly\r\n\r\npublic inline fun
<T> Comparator<T>.thenByDescending(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =\r\n
Comparator { a, b ->\r\n    val previousCompare = this@thenByDescending.compare(a, b)\r\n    if
(previousCompare != 0) previousCompare else compareValuesBy(b, a, selector)\r\n    }\r\n\r\n/**\r\n * Creates a
descending comparator comparing values after the primary comparator defined them equal. It uses\r\n * the
[selector] function to transform values and then compares them with the given [comparator].\r\n * \r\n * @sample
samples.comparisons.Comparisons.thenByDescendingWithComparator\r\n
* \r\n * @kotlin.internal.InlineOnly\r\n\r\npublic inline fun <T, K> Comparator<T>.thenByDescending(comparator:
Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =\r\n
Comparator { a, b ->\r\n    val
previousCompare = this@thenByDescending.compare(a, b)\r\n    if (previousCompare != 0) previousCompare
else compareValuesBy(b, a, comparator, selector)\r\n    }\r\n\r\n\r\n/**\r\n * Creates a comparator using the primary
comparator and function to calculate a result of comparison.\r\n * \r\n * @sample
samples.comparisons.Comparisons.thenComparator\r\n
* \r\n * @kotlin.internal.InlineOnly\r\n\r\npublic inline fun <T>
Comparator<T>.thenComparator(crossinline comparison: (a: T, b: T) -> Int): Comparator<T> =\r\n
Comparator {
a, b ->\r\n    val previousCompare = this@thenComparator.compare(a, b)\r\n    if (previousCompare != 0)
previousCompare else comparison(a, b)\r\n    }\r\n\r\n\r\n/**\r\n * Combines this comparator and the given
[comparator] such that the latter is applied only\r\n * when the former considered values equal.\r\n * \r\n * @sample
samples.comparisons.Comparisons.then\r\n
* \r\n * @kotlin.internal.InlineOnly\r\n\r\npublic infix fun <T> Comparator<T>.then(comparator:
Comparator<in T>): Comparator<T> =\r\n
Comparator { a, b ->\r\n    val previousCompare =
this@then.compare(a, b)\r\n    if (previousCompare != 0) previousCompare else comparator.compare(a, b)\r\n
}\r\n\r\n\r\n/**\r\n * Combines this comparator and the given [comparator] such that the latter is applied only\r\n *
when the former considered values equal.\r\n * \r\n * @sample
samples.comparisons.Comparisons.thenDescending\r\n
* \r\n * @kotlin.internal.InlineOnly\r\n\r\npublic infix fun <T>
Comparator<T>.thenDescending(comparator: Comparator<in T>): Comparator<T> =\r\n
Comparator<T> { a, b -
>\r\n    val previousCompare = this@thenDescending.compare(a, b)\r\n    if (previousCompare != 0)
previousCompare else comparator.compare(b, a)\r\n    }\r\n\r\n\r\n// Not so useful without type inference for receiver of
expression\r\n\r\n/**\r\n * Extends the given [comparator] of non-nullable values to a comparator of nullable values\r\n
* considering `null` value less than any other value.\r\n * \r\n * @sample
samples.comparisons.Comparisons.nullsFirstLastWithComparator\r\n
* \r\n * @kotlin.internal.InlineOnly\r\n\r\npublic fun <T : Any>
nullsFirst(comparator: Comparator<in T>): Comparator<T?> =\r\n
Comparator { a, b ->\r\n    when {\r\n
a === b -> 0\r\n    a == null -> -1\r\n    b == null -> 1\r\n    else -> comparator.compare(a, b)\r\n
}\r\n    }\r\n\r\n\r\n/**\r\n * Provides a comparator of nullable [Comparable] values\r\n * considering `null` value less
than any other value.\r\n * \r\n * @sample samples.comparisons.Comparisons.nullsFirstLastComparator\r\n
* \r\n * @kotlin.internal.InlineOnly\r\n\r\npublic inline fun <T : Comparable<T>> nullsFirst(): Comparator<T?> =
nullsFirst(naturalOrder())\r\n\r\n\r\n/**\r\n * Extends the given [comparator] of non-nullable values to a comparator of
nullable values\r\n * considering `null` value greater than any other value.\r\n * \r\n * @sample
samples.comparisons.Comparisons.nullsFirstLastWithComparator\r\n
* \r\n * @kotlin.internal.InlineOnly\r\n\r\npublic fun <T : Any>
nullsLast(comparator: Comparator<in T>): Comparator<T?> =\r\n
Comparator { a, b ->\r\n    when {\r\n
a === b -> 0\r\n    a == null -> 1\r\n    b == null -> -1\r\n    else -> comparator.compare(a, b)\r\n
}\r\n    }\r\n\r\n\r\n/**\r\n * Provides a comparator of nullable [Comparable] values\r\n * considering `null` value
greater than any other value.\r\n * \r\n * @sample samples.comparisons.Comparisons.nullsFirstLastComparator\r\n
* \r\n * @kotlin.internal.InlineOnly\r\n\r\npublic inline fun <T : Comparable<T>> nullsLast(): Comparator<T?> =
nullsLast(naturalOrder())\r\n\r\n\r\n/**\r\n * Returns a comparator that compares [Comparable] objects in natural
order.\r\n * \r\n * @sample samples.comparisons.Comparisons.naturalOrderComparator\r\n
* \r\n * @kotlin.internal.InlineOnly\r\n\r\npublic fun <T :
Comparable<T>> naturalOrder(): Comparator<T> = @Suppress("UNCHECKED_CAST")

```

```

(NaturalOrderComparator as Comparator<T>)\r\n\r\n/**\r\n * Returns a comparator that compares [Comparable]
objects in reversed natural order.\r\n * @sample
samples.comparisons.Comparisons.nullsFirstLastWithComparator\r\n * \r\npublic fun <T : Comparable<T>>
reverseOrder(): Comparator<T> = @Suppress("UNCHECKED_CAST") (ReverseOrderComparator as
Comparator<T>)\r\n\r\n/**\r\n * Returns a comparator that imposes the reverse ordering of this comparator.\r\n
*\r\n * @sample samples.comparisons.Comparisons.reversed\r\n * \r\npublic fun <T> Comparator<T>.reversed():
Comparator<T> = when (this) {\r\n    is ReversedComparator -> this.comparator\r\n    NaturalOrderComparator ->
@Suppress("UNCHECKED_CAST") (ReverseOrderComparator as Comparator<T>)\r\n
ReverseOrderComparator -> @Suppress("UNCHECKED_CAST") (NaturalOrderComparator as
Comparator<T>)\r\n    else -> ReversedComparator(this)\r\n}\r\n\r\n\r\nprivate class
ReversedComparator<T>(public val comparator: Comparator<T>) : Comparator<T> {\r\n    override fun compare(a:
T, b: T): Int = comparator.compare(b, a)\r\n    @Suppress("VIRTUAL_MEMBER_HIDDEN")\r\n    fun
reversed(): Comparator<T> = comparator\r\n}\r\n\r\nprivate object NaturalOrderComparator :
Comparator<Comparable<Any>> {\r\n    override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int =
a.compareTo(b)\r\n    @Suppress("VIRTUAL_MEMBER_HIDDEN")\r\n    fun reversed():
Comparator<Comparable<Any>> = ReverseOrderComparator\r\n}\r\n\r\nprivate object ReverseOrderComparator :
Comparator<Comparable<Any>> {\r\n    override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int =
b.compareTo(a)\r\n    @Suppress("VIRTUAL_MEMBER_HIDDEN")\r\n    fun reversed():
Comparator<Comparable<Any>> = NaturalOrderComparator\r\n}\r\n\r\n"/**\r\n * Copyright 2010-2018 JetBrains
s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0
license that can be found in the license/LICENSE.txt file.\r\n * \r\n\r\npackage kotlin\r\n\r\n\r\npublic actual
interface Comparator<T> {\r\n    @JsName("compare")\r\n    actual fun compare(a: T, b: T): Int\r\n}\r\n\r\n\r\npublic
actual inline fun <T> Comparator(crossinline comparison: (a: T, b: T) -> Int): Comparator<T> = object :
Comparator<T> {\r\n    override fun compare(a: T, b: T): Int = comparison(a, b)\r\n}\r\n\r\n"/**\r\n * Copyright 2010-
2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by
the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*\r\n\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n\r\n@file:kotlin.jvm.JvmName("StandardKt")\r\n\r\npackage
kotlin\r\n\r\n\r\nimport kotlin.contracts.*\r\n\r\n\r\n/**\r\n * An exception is thrown to indicate that a method body remains
to be implemented.\r\n * \r\n\r\npublic class NotImplementedError(message: String = "An operation is not
implemented.") : Error(message)\r\n\r\n\r\n/**\r\n * Always throws [NotImplementedError] stating that operation is
not implemented.\r\n * \r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun TODO(): Nothing = throw
NotImplementedError()\r\n\r\n\r\n/**\r\n * Always throws [NotImplementedError] stating that operation is not
implemented.\r\n * \r\n * @param reason a string explaining why the implementation is missing.\r\n
*\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun TODO(reason: String): Nothing = throw
NotImplementedError("An operation is not implemented: $reason")\r\n\r\n\r\n\r\n\r\n\r\n/**\r\n * Calls the specified
function [block] and returns its result.\r\n * \r\n * For detailed usage information see the documentation for [scope
functions](https://kotlinlang.org/docs/reference/scope-functions.html#run).\r\n
*\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun <R> run(block: () -> R): R {\r\n    contract {\r\n
callsInPlace(block, InvocationKind.EXACTLY_ONCE)\r\n    }\r\n    return block()\r\n}\r\n\r\n\r\n\r\n/**\r\n * Calls the
specified function [block] with `this` value as its receiver and returns its result.\r\n * \r\n * For detailed usage
information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-
functions.html#run).\r\n
*\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun <T, R> T.run(block: T.() -> R): R
{\r\n    contract {\r\n        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\r\n    }\r\n    return
block()\r\n}\r\n\r\n\r\n\r\n\r\n\r\n/**\r\n * Calls the specified function [block] with the given [receiver] as its receiver and returns
its result.\r\n * \r\n * For detailed usage information see the documentation for [scope
functions](https://kotlinlang.org/docs/reference/scope-functions.html#with).\r\n
*\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun <T, R> with(receiver: T, block: T.() -> R): R {\r\n    contract
{\r\n        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\r\n    }\r\n    return

```

```

receiver.block()\r\n}\r\n\r\n/**\r\n * Calls the specified function [block] with `this` value as its receiver and returns
`this` value.\r\n *\r\n * For detailed usage information see the documentation for [scope
functions](https://kotlinlang.org/docs/reference/scope-functions.html#apply).\r\n
*\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T> T.apply(block: T.() -> Unit): T {\r\n  contract {\r\n
callsInPlace(block, InvocationKind.EXACTLY_ONCE)\r\n  }\r\n  block()\r\n  return this\r\n}\r\n\r\n/**\r\n *
Calls the specified function [block] with `this` value as its argument and returns `this` value.\r\n *\r\n * For detailed
usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-
functions.html#also).\r\n *\r\n@kotlin.internal.InlineOnly\r\n@SinceKotlin("1.1")\r\npublic inline fun <T>
T.also(block: (T) -> Unit): T {\r\n  contract {\r\n    callsInPlace(block, InvocationKind.EXACTLY_ONCE)\r\n
}\r\n  block(this)\r\n  return this\r\n}\r\n\r\n/**\r\n * Calls the specified function [block] with `this` value as its
argument and returns its result.\r\n *\r\n * For detailed usage information see the documentation for [scope
functions](https://kotlinlang.org/docs/reference/scope-functions.html#let).\r\n
*\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T, R> T.let(block: (T) -> R): R {\r\n  contract {\r\n
callsInPlace(block, InvocationKind.EXACTLY_ONCE)\r\n  }\r\n  return block(this)\r\n}\r\n\r\n/**\r\n * Returns
`this` value if it satisfies the given [predicate] or `null`, if it doesn't.\r\n
*\r\n@kotlin.internal.InlineOnly\r\n@SinceKotlin("1.1")\r\npublic inline fun <T> T.takeIf(predicate: (T) ->
Boolean): T? {\r\n  contract {\r\n    callsInPlace(predicate, InvocationKind.EXACTLY_ONCE)\r\n  }\r\n
return if (predicate(this)) this else null\r\n}\r\n\r\n/**\r\n * Returns `this` value if it _does not_ satisfy the given
[predicate] or `null`, if it does.\r\n *\r\n@kotlin.internal.InlineOnly\r\n@SinceKotlin("1.1")\r\npublic inline fun
<T> T.takeUnless(predicate: (T) -> Boolean): T? {\r\n  contract {\r\n    callsInPlace(predicate,
InvocationKind.EXACTLY_ONCE)\r\n  }\r\n  return if (!predicate(this)) this else null\r\n}\r\n\r\n/**\r\n *
Executes the given function [action] specified number of [times].\r\n *\r\n * A zero-based index of current iteration
is passed as a parameter to [action].\r\n *\r\n * @sample samples.misc.ControlFlow.repeat\r\n
*\r\n@kotlin.internal.InlineOnly\r\npublic inline fun repeat(times: Int, action: (Int) -> Unit) {\r\n  contract {\r\n
callsInPlace(action) }\r\n\r\n  for (index in 0 until times) {\r\n    action(index)\r\n  }\r\n}\r\n\r\n", /*\n * Copyright
2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed
by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\npackage
kotlin.comparisons\n\n/\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n\n/\n\nimport kotlin.js.*\n\n/**\n * Returns the greater
of two values.\n * If values are equal, returns the first one.\n *\n@SinceKotlin("1.1")\npublic actual fun <T :
Comparable<T>> maxOf(a: T, b: T): T {\n  return if (a >= b) a else b\n}\n\n/**\n * Returns the greater of two
values.\n
*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun maxOf(a: Byte, b: Byte): Byte {\n  return Math.max(a.toInt(), b.toInt()).unsafeCast<Byte>()\n}\n\n/**\n *
Returns the greater of two values.\n
*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun maxOf(a: Short, b: Short): Short {\n  return Math.max(a.toInt(),
b.toInt()).unsafeCast<Short>()\n}\n\n/**\n * Returns the greater of two values.\n
*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun maxOf(a: Int, b: Int): Int {\n  return Math.max(a, b)\n}\n\n/**\n * Returns the greater of two
values.\n
*\n@SinceKotlin("1.1")\n@Suppress("DEPRECATION_ERROR", "NOTHING_TO_INLINE")\npublic actual
inline fun maxOf(a: Long, b: Long): Long {\n  return if (a >= b) a else b\n}\n\n/**\n * Returns the greater of two
values.\n
*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun maxOf(a: Float, b: Float): Float {\n  return Math.max(a, b)\n}\n\n/**\n * Returns the greater of two
values.\n
*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun maxOf(a: Double, b: Double): Double {\n  return Math.max(a, b)\n}\n\n/**\n * Returns the greater of

```

```

three values.\n *\n@SinceKotlin("1.1")\npublic actual fun <T : Comparable<T>> maxOf(a: T, b: T, c: T): T {\n
return maxOf(a, maxOf(b, c))\n}\n\n/**\n * Returns the greater of three values.\n
*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun maxOf(a: Byte, b: Byte, c: Byte): Byte {\n  return Math.max(a.toInt(), b.toInt(),
c.toInt()).unsafeCast<Byte>()\n}\n\n/**\n * Returns the greater of three values.\n
*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun maxOf(a: Short, b: Short, c: Short): Short {\n  return Math.max(a.toInt(), b.toInt(),
c.toInt()).unsafeCast<Short>()\n}\n\n/**\n * Returns the greater of three values.\n
*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun maxOf(a: Int, b: Int, c: Int): Int {\n  return Math.max(a, b, c)\n}\n\n/**\n * Returns the greater of three
values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic actual inline fun maxOf(a: Long, b: Long,
c: Long): Long {\n  return maxOf(a, maxOf(b, c))\n}\n\n/**\n * Returns the greater of three values.\n
*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun maxOf(a: Float, b: Float, c: Float): Float {\n  return Math.max(a, b, c)\n}\n\n/**\n * Returns the greater
of three values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun maxOf(a: Double, b: Double, c: Double): Double {\n  return Math.max(a, b, c)\n}\n\n/**\n * Returns the
smaller of two values.\n *\n If values are equal, returns the first one.\n *\n@SinceKotlin("1.1")\npublic actual fun
<T : Comparable<T>> minOf(a: T, b: T): T {\n  return if (a <= b) a else b\n}\n\n/**\n * Returns the smaller of two
values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Byte, b: Byte): Byte {\n  return Math.min(a.toInt(), b.toInt()).unsafeCast<Byte>()\n}\n\n/**\n
* Returns the smaller of two values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Short, b: Short): Short {\n  return Math.min(a.toInt(),
b.toInt()).unsafeCast<Short>()\n}\n\n/**\n * Returns the smaller of two values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Int, b: Int): Int {\n  return Math.min(a, b)\n}\n\n/**\n * Returns the smaller of two values.\n
*\n@SinceKotlin("1.1")\n@Suppress("DEPRECATION_ERROR", "NOTHING_TO_INLINE")\npublic actual
inline fun minOf(a: Long, b: Long): Long {\n  return if (a <= b) a else b\n}\n\n/**\n * Returns the smaller of two
values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Float, b: Float): Float {\n  return Math.min(a, b)\n}\n\n/**\n * Returns the smaller of two
values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Double, b: Double): Double {\n  return Math.min(a, b)\n}\n\n/**\n * Returns the smaller of
three values.\n *\n@SinceKotlin("1.1")\npublic actual fun <T : Comparable<T>> minOf(a: T, b: T, c: T): T {\n
return minOf(a, minOf(b, c))\n}\n\n/**\n * Returns the smaller of three values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Byte, b: Byte, c: Byte): Byte {\n  return Math.min(a.toInt(), b.toInt(),
c.toInt()).unsafeCast<Byte>()\n}\n\n/**\n * Returns the smaller of three values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Short, b: Short, c: Short): Short {\n  return Math.min(a.toInt(), b.toInt(),
c.toInt()).unsafeCast<Short>()\n}\n\n/**\n * Returns the smaller of three values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Int, b: Int, c: Int): Int {\n  return Math.min(a, b, c)\n}\n\n/**\n * Returns the smaller of three
values.\n *\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Long, b: Long,
c: Long): Long {\n  return minOf(a, minOf(b, c))\n}\n\n/**\n * Returns the smaller of three values.\n

```

```

*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Float, b: Float, c: Float): Float {\n    return Math.min(a, b, c)\n}\n\n/**\n * Returns the smaller
of three values.\n
*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\n@Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Double, b: Double, c: Double): Double {\n    return Math.min(a, b, c)\n}\n\n"/*\n * Copyright
2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed
by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("CollectionsKt")\n\npackage
kotlin.collections\n\n/\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n/\n\nimport kotlin.random.*\nimport
kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns 1st *element* from the collection.\n
*\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> List<T>.component1(): T {\n    return
get(0)\n}\n\n/**\n * Returns 2nd *element* from the collection.\n *\n@kotlin.internal.InlineOnly\npublic inline
operator fun <T> List<T>.component2(): T {\n    return get(1)\n}\n\n/**\n * Returns 3rd *element* from the
collection.\n *\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> List<T>.component3(): T {\n    return
get(2)\n}\n\n/**\n * Returns 4th *element* from the collection.\n *\n@kotlin.internal.InlineOnly\npublic inline
operator fun <T> List<T>.component4(): T {\n    return get(3)\n}\n\n/**\n * Returns 5th *element* from the
collection.\n *\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> List<T>.component5(): T {\n    return
get(4)\n}\n\n/**\n * Returns `true` if [element] is found in the collection.\n *\npublic operator fun
<@kotlin.internal.OnlyInputTypes T> Iterable<T>.contains(element: T): Boolean {\n    if (this is Collection)\n    return contains(element)\n    return indexOf(element) >= 0\n}\n\n/**\n * Returns an element at the given [index] or
throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this collection.\n * \n * @sample
samples.collections.Collections.Elements.elementAt\n *\npublic fun <T> Iterable<T>.elementAt(index: Int): T {\n
if (this is List)\n    return get(index)\n    return elementAtOrElse(index) { throw
IndexOutOfBoundsException("Collection doesn't contain element at index $index.") }\n}\n\n/**\n * Returns an
element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this list.\n
* \n * @sample samples.collections.Collections.Elements.elementAt\n *\n@kotlin.internal.InlineOnly\npublic
inline fun <T> List<T>.elementAt(index: Int): T {\n    return get(index)\n}\n\n/**\n * Returns an element at the
given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this collection.\n
* \n * @sample samples.collections.Collections.Elements.elementAtOrElse\n *\npublic fun <T>
Iterable<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {\n    if (this is List)\n    return
this.getOrElse(index, defaultValue)\n    if (index < 0)\n    return defaultValue(index)\n    val iterator = iterator()\n
var count = 0\n    while (iterator.hasNext()) {\n        val element = iterator.next()\n        if (index == count++)\n        return element\n    }\n    return defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the
result of calling the [defaultValue] function if the [index] is out of bounds of this list.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n *\n@kotlin.internal.InlineOnly\npublic inline fun
<T> List<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {\n    return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the
[index] is out of bounds of this collection.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n *\npublic fun <T>
Iterable<T>.elementAtOrNull(index: Int): T? {\n    if (this is List)\n    return this.elementAtOrNull(index)\n    if (index <
0)\n    return null\n    val iterator = iterator()\n    var count = 0\n    while (iterator.hasNext()) {\n        val element =
iterator.next()\n        if (index == count++)\n        return element\n    }\n    return null\n}\n\n/**\n * Returns an
element at the given [index] or `null` if the [index] is out of bounds of this list.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n *\n@kotlin.internal.InlineOnly\npublic inline fun
<T> List<T>.elementAtOrNull(index: Int): T? {\n    return this.elementAtOrNull(index)\n}\n\n/**\n * Returns the first
element matching the given [predicate], or `null` if no such element was found.\n
*\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.find(predicate: (T) -> Boolean): T? {\n    return

```



```

firstOrNull(predicate)\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such
element was found.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.findLast(predicate: (T) ->
Boolean): T? {\n    return lastOrNull(predicate)\n}\n\n/**\n * Returns the last element matching the given
[predicate], or `null` if no such element was found.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T>
List<T>.findLast(predicate: (T) -> Boolean): T? {\n    return lastOrNull(predicate)\n}\n\n/**\n * Returns first
element.\n * @throws [NoSuchElementException] if the collection is empty.\n */\npublic fun <T>
Iterable<T>.first(): T {\n    when (this) {\n        is List -> return this.first()\n        else -> {\n            val iterator =
iterator()\n            if (!iterator.hasNext())\n                throw NoSuchElementException("Collection is empty.")\n            return iterator.next()\n        }\n    }\n}\n\n/**\n * Returns first element.\n * @throws [NoSuchElementException]
if the list is empty.\n */\npublic fun <T> List<T>.first(): T {\n    if (isEmpty())\n        throw
NoSuchElementException("List is empty.")\n    return this[0]\n}\n\n/**\n * Returns the first element matching the
given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun <T>
Iterable<T>.first(predicate: (T) -> Boolean): T {\n    for (element in this) if (predicate(element)) return element\n
throw NoSuchElementException("Collection contains no element matching the predicate.")\n}\n\n/**\n * Returns
the first element, or `null` if the collection is empty.\n */\npublic fun <T> Iterable<T>.firstOrNull(): T? {\n    when
(this) {\n        is List -> {\n            if (isEmpty())\n                return null\n            else\n                return this[0]\n        }\n        else -> {\n            val iterator = iterator()\n            if (!iterator.hasNext())\n                return null\n            return iterator.next()\n        }\n    }\n}\n\n/**\n * Returns the first element
matching the given [predicate], or `null` if element was not found.\n */\npublic inline fun <T>
Iterable<T>.firstOrNull(predicate: (T) -> Boolean): T? {\n    for (element in this) if (predicate(element)) return
element\n    return null\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the
[defaultValue] function if the [index] is out of bounds of this list.\n */\n@kotlin.internal.InlineOnly\npublic inline
fun <T> List<T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {\n    return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the
[index] is out of bounds of this list.\n */\npublic fun <T> List<T>.getOrElse(index: Int): T? {\n    return if (index >=
0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns first index of [element], or -1 if the collection
does not contain element.\n */\npublic fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.indexOf(element: T):
Int {\n    if (this is List) return this.indexOf(element)\n    var index = 0\n    for (item in this) {\n
checkIndexOverflow(index)\n        if (element == item)\n            return index\n        index++\n    }\n    return -
1\n}\n\n/**\n * Returns first index of [element], or -1 if the list does not contain element.\n */\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in
some cases\npublic fun <@kotlin.internal.OnlyInputTypes T> List<T>.indexOf(element: T): Int {\n    return
indexOf(element)\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the
collection does not contain such element.\n */\npublic inline fun <T> Iterable<T>.indexOfFirst(predicate: (T) ->
Boolean): Int {\n    var index = 0\n    for (item in this) {\n        checkIndexOverflow(index)\n        if
(predicate(item))\n            return index\n        index++\n    }\n    return -1\n}\n\n/**\n * Returns index of the first
element matching the given [predicate], or -1 if the list does not contain such element.\n */\npublic inline fun <T>
List<T>.indexOfFirst(predicate: (T) -> Boolean): Int {\n    var index = 0\n    for (item in this) {\n        if
(predicate(item))\n            return index\n        index++\n    }\n    return -1\n}\n\n/**\n * Returns index of the last
element matching the given [predicate], or -1 if the collection does not contain such element.\n */\npublic inline fun
<T> Iterable<T>.indexOfLast(predicate: (T) -> Boolean): Int {\n    var lastIndex = -1\n    var index = 0\n    for (item
in this) {\n        checkIndexOverflow(index)\n        if (predicate(item))\n            lastIndex = index\n        index++\n
    }\n    return lastIndex\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the list
does not contain such element.\n */\npublic inline fun <T> List<T>.indexOfLast(predicate: (T) -> Boolean): Int {\n
val iterator = this.listIterator(size)\n    while (iterator.hasPrevious()) {\n        if (predicate(iterator.previous())) {\n
return iterator.nextIndex()\n        }\n    }\n    return -1\n}\n\n/**\n * Returns the last element.\n * @throws
[NoSuchElementException] if the collection is empty.\n */\npublic fun <T> Iterable<T>.last(): T {\n    when (this)

```

```

{\n    is List -> return this.last()\n    else -> {\n        val iterator = iterator()\n        if (!iterator.hasNext())\n            throw NoSuchElementException("Collection is empty.")\n        var last = iterator.next()\n        while\n            (iterator.hasNext())\n                last = iterator.next()\n        return last\n    }\n}\n\n/**\n * Returns the last\n element.\n * @throws [NoSuchElementException] if the list is empty.\n */\npublic fun <T> List<T>.last(): T {\n    if (isEmpty())\n        throw NoSuchElementException("List is empty.")\n    return this[lastIndex]\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws [NoSuchElementException] if no such\n element is found.\n */\npublic inline fun <T> Iterable<T>.last(predicate: (T) -> Boolean): T {\n    var last: T? =\n        null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            last = element\n            found = true\n        }\n    }\n    if (!found) throw NoSuchElementException("Collection contains no element\n matching the predicate.")\n    @Suppress("UNCHECKED_CAST")\n    return last as T\n}\n\n/**\n * Returns the\n last element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\npublic inline fun <T> List<T>.last(predicate: (T) -> Boolean): T {\n    val iterator = this.listIterator(size)\n    while (iterator.hasPrevious()) {\n        val element = iterator.previous()\n        if (predicate(element)) return\n            element\n    }\n    throw NoSuchElementException("List contains no element matching the predicate.")\n}\n\n/**\n * Returns last index of [element], or -1 if the collection does not contain element.\n */\npublic fun\n<@kotlin.internal.OnlyInputTypes T> Iterable<T>.lastIndexOf(element: T): Int {\n    if (this is List) return\n        this.lastIndexOf(element)\n    var lastIndex = -1\n    var index = 0\n    for (item in this) {\n        checkIndexOverflow(index)\n        if (element == item)\n            lastIndex = index\n            index++\n    }\n    return\n        lastIndex\n}\n\n/**\n * Returns last index of [element], or -1 if the list does not contain element.\n */\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in\n some cases\npublic fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {\n    return\n        lastIndexOf(element)\n}\n\n/**\n * Returns the last element, or `null` if the collection is empty.\n */\npublic fun\n<T> Iterable<T>.lastOrNull(): T? {\n    when (this) {\n        is List -> return if (isEmpty()) null else this[size - 1]\n        else -> {\n            val iterator = iterator()\n            if (iterator.hasNext())\n                return null\n            var last =\n                iterator.next()\n            while (iterator.hasNext())\n                last = iterator.next()\n            return last\n        }\n    }\n}\n\n/**\n * Returns the last element, or `null` if the list is empty.\n */\npublic fun <T> List<T>.lastOrNull(): T?\n {\n    return if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element matching the given\n [predicate], or `null` if no such element was found.\n */\npublic inline fun <T> Iterable<T>.lastOrNull(predicate:\n (T) -> Boolean): T? {\n    var last: T? = null\n    for (element in this) {\n        if (predicate(element)) {\n            last =\n                element\n        }\n    }\n    return last\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null`\n if no such element was found.\n */\npublic inline fun <T> List<T>.lastOrNull(predicate: (T) -> Boolean): T? {\n    val iterator = this.listIterator(size)\n    while (iterator.hasPrevious()) {\n        val element = iterator.previous()\n        if (predicate(element)) return element\n    }\n    return null\n}\n\n/**\n * Returns a random element from this\n collection.\n * @throws NoSuchElementException if this collection is empty.\n */\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Collection<T>.random(): T {\n    return random(Random)\n}\n\n/**\n * Returns a random element from this collection using the specified source of\n randomness.\n * @throws NoSuchElementException if this collection is empty.\n */\n@SinceKotlin("1.3")\npublic fun <T> Collection<T>.random(random: Random): T {\n    if (isEmpty())\n        throw NoSuchElementException("Collection is empty.")\n    return elementAt(random.nextInt(size))\n}\n\n/**\n * Returns the single element, or throws an exception if the collection is empty or has more than one element.\n */\npublic fun <T> Iterable<T>.single(): T {\n    when (this) {\n        is List -> return this.single()\n        else -> {\n            val iterator = iterator()\n            if (iterator.hasNext())\n                throw NoSuchElementException("Collection\n is empty.")\n            val single = iterator.next()\n            if (iterator.hasNext())\n                throw\n                    IllegalArgumentException("Collection has more than one element.")\n            return single\n        }\n    }\n}\n\n/**\n * Returns the single element, or throws an exception if the list is empty or has more than one\n element.\n */\npublic fun <T> List<T>.single(): T {\n    return when (size) {\n        0 -> throw\n            NoSuchElementException("List is empty.")\n        1 -> this[0]\n        else -> throw\n            IllegalArgumentException("List has more than one element.")\n    }\n}\n\n/**\n * Returns the single element

```

```

matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic
inline fun <T> Iterable<T>.single(predicate: (T) -> Boolean): T {\n    var single: T? = null\n    var found = false\n
for (element in this) {\n    if (predicate(element)) {\n        if (found) throw
IllegalArgumentOutOfRangeException("Collection contains more than one matching element.")\n        single = element\n
found = true\n    } } if (!found) throw NoSuchElementException("Collection contains no element
matching the predicate.")\n    @Suppress("UNCHECKED_CAST")\n    return single as T\n}\n\n/**\n * Returns
single element, or `null` if the collection is empty or has more than one element.\n */\npublic fun <T>
Iterable<T>.singleOrNull(): T? {\n    when (this) {\n        is List -> return if (size == 1) this[0] else null\n    else ->
{\n        val iterator = iterator()\n        if (!iterator.hasNext())\n            return null\n        val single =
iterator.next()\n        if (iterator.hasNext())\n            return null\n        return single\n    } }\n}\n\n/**\n
* Returns single element, or `null` if the list is empty or has more than one element.\n */\npublic fun <T>
List<T>.singleOrNull(): T? {\n    return if (size == 1) this[0] else null\n}\n\n/**\n * Returns the single element
matching the given [predicate], or `null` if element was not found or more than one element was found.\n */\npublic
inline fun <T> Iterable<T>.singleOrNull(predicate: (T) -> Boolean): T? {\n    var single: T? = null\n    var found =
false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single =
element\n            found = true\n        } } if (!found) return null\n    return single\n}\n\n/**\n * Returns a list
containing all elements except first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n *
\n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun <T> Iterable<T>.drop(n: Int):
List<T> {\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return toList()\n
val list: ArrayList<T>\n    if (this is Collection<*>) {\n        val resultSize = size - n\n        if (resultSize <= 0)\n
return emptyList()\n        if (resultSize == 1)\n            return listOf(last())\n        list = ArrayList<T>(resultSize)\n
if (this is List<T>) {\n            if (this is RandomAccess) {\n                for (index in n until size)\n
list.add(this[index])\n            } else {\n                for (item in listIterator(n))\n                    list.add(item)\n            }\n
return list\n        } } else {\n            list = ArrayList<T>()\n            var count = 0\n            for (item in this) {\n
if (count >= n) list.add(item) else ++count\n        }\n        return list.optimizeReadOnlyList()\n}\n\n/**\n * Returns a list
containing all elements except last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n *
\n * @sample samples.collections.Collections.Transformations.drop\n */\npublic fun <T> List<T>.dropLast(n: Int):
List<T> {\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    return take((size -
n).coerceAtLeast(0))\n}\n\n/**\n * Returns a list containing all elements except last elements that satisfy the given
[predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n */\npublic inline fun <T>
List<T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {\n    if (!isEmpty()) {\n        val iterator =
listIterator(size)\n        while (iterator.hasPrevious()) {\n            if (!predicate(iterator.previous())) {\n                return
take(iterator.nextIndex() + 1)\n            } }\n    }\n    return emptyList()\n}\n\n/**\n * Returns a list containing
all elements except first elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n */\npublic inline fun <T> Iterable<T>.dropWhile(predicate:
(T) -> Boolean): List<T> {\n    var yielding = false\n    val list = ArrayList<T>()\n    for (item in this)\n        if
(yielding)\n            list.add(item)\n        else if (!predicate(item)) {\n            list.add(item)\n            yielding = true\n
        }\n    return list\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n */\npublic
inline fun <T> Iterable<T>.filter(predicate: (T) -> Boolean): List<T> {\n    return filterTo(ArrayList<T>(),
predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param
[predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate
evaluation on the element.\n */\npublic inline fun <T> Iterable<T>.filterIndexed(predicate: (index: Int, T) ->
Boolean): List<T> {\n    return filterIndexedTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Appends all elements
matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an
element and the element itself\n * and returns the result of predicate evaluation on the element.\n */\npublic inline
fun <T, C : MutableCollection<in T>> Iterable<T>.filterIndexedTo(destination: C, predicate: (index: Int, T) ->
Boolean): C {\n    forEachIndexed { index, element ->\n        if (predicate(index, element))\n            destination.add(element)\n    }\n    return destination\n}\n\n/**\n * Returns a list containing all elements that are

```

```

instances of specified type parameter R.\n */\npublic inline fun <reified R> Iterable<*>.filterIsInstance():
List<@kotlin.internal.NoInfer R> {\n    return filterIsInstanceTo(ArrayList<R>())\n}\n\n/**\n * Appends all
elements that are instances of specified type parameter R to the given [destination].\n */\npublic inline fun <reified
R, C : MutableCollection<in R>> Iterable<*>.filterIsInstanceTo(destination: C): C {\n    for (element in this) if
(element is R) destination.add(element)\n    return destination\n}\n\n/**\n * Returns a list containing all elements
not matching the given [predicate].\n */\npublic inline fun <T> Iterable<T>.filterNot(predicate: (T) -> Boolean):
List<T> {\n    return filterNotTo(ArrayList<T>(), predicate)\n}\n\n/**\n * Returns a list containing all elements that
are not `null`.\n */\npublic fun <T : Any> Iterable<T?>.filterNotNull(): List<T> {\n    return
filterNotNullTo(ArrayList<T>())\n}\n\n/**\n * Appends all elements that are not `null` to the given [destination].\n
*/\npublic fun <C : MutableCollection<in T>, T : Any> Iterable<T?>.filterNotNullTo(destination: C): C {\n    for
(element in this) if (element != null) destination.add(element)\n    return destination\n}\n\n/**\n * Appends all
elements not matching the given [predicate] to the given [destination].\n */\npublic inline fun <T, C :
MutableCollection<in T>> Iterable<T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {\n    for (element
in this) if (!predicate(element)) destination.add(element)\n    return destination\n}\n\n/**\n * Appends all elements
matching the given [predicate] to the given [destination].\n */\npublic inline fun <T, C : MutableCollection<in T>>
Iterable<T>.filterTo(destination: C, predicate: (T) -> Boolean): C {\n    for (element in this) if (predicate(element))
destination.add(element)\n    return destination\n}\n\n/**\n * Returns a list containing elements at indices in the
specified [indices] range.\n */\npublic fun <T> List<T>.slice(indices: IntRange): List<T> {\n    if
(indices.isEmpty()) return listOf()\n    return this.subList(indices.start, indices.endInclusive + 1).toList()\n}\n\n/**\n
* Returns a list containing elements at specified [indices].\n */\npublic fun <T> List<T>.slice(indices:
Iterable<Int>): List<T> {\n    val size = indices.collectionSizeOrDefault(10)\n    if (size == 0) return emptyList()\n
val list = ArrayList<T>(size)\n    for (index in indices) {\n        list.add(get(index))\n    }\n    return list\n}\n\n/**\n
* Returns a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n *
@sample samples.collections.Collections.Transformations.take\n */\npublic fun <T> Iterable<T>.take(n: Int):
List<T> {\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return
emptyList()\n    if (this is Collection<T>) {\n        if (n >= size) return toList()\n        if (n == 1) return
listOf(first())\n    }\n    var count = 0\n    val list = ArrayList<T>(n)\n    for (item in this) {\n        if (count++ == n)\n            break\n        list.add(item)\n    }\n    return list.optimizeReadOnlyList()\n}\n\n/**\n * Returns a list containing
last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n */\npublic fun <T> List<T>.takeLast(n: Int): List<T> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return emptyList()\n    val size =
size\n    if (n >= size) return toList()\n    if (n == 1) return listOf(last())\n    val list = ArrayList<T>(n)\n    if (this
is RandomAccess) {\n        for (index in size - n until size)\n            list.add(this[index])\n    } else {\n        for (item in
listIterator(size - n))\n            list.add(item)\n    }\n    return list\n}\n\n/**\n * Returns a list containing last
elements satisfying the given [predicate].\n * \n * @sample samples.collections.Collections.Transformations.take\n */\npublic
inline fun <T> List<T>.takeLastWhile(predicate: (T) -> Boolean): List<T> {\n    if (isEmpty())\n        return
emptyList()\n    val iterator = listIterator(size)\n    while (iterator.hasPrevious()) {\n        if
(!predicate(iterator.previous())) {\n            iterator.next()\n            val expectedSize = size - iterator.nextIndex()\n
            if (expectedSize == 0) return emptyList()\n            return ArrayList<T>(expectedSize).apply {\n                while
(iterator.hasNext())\n                    add(iterator.next())\n            }\n        }\n    }\n    return toList()\n}\n\n/**\n
* Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n */\npublic inline fun <T> Iterable<T>.takeWhile(predicate:
(T) -> Boolean): List<T> {\n    val list = ArrayList<T>()\n    for (item in this) {\n        if (!predicate(item))\n            break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n * Reverses elements in the list in-place.\n */\npublic
expect fun <T> MutableList<T>.reverse(): Unit\n\n/**\n * Returns a list with elements in reversed order.\n */\npublic fun <T>
Iterable<T>.reversed(): List<T> {\n    if (this is Collection && size <= 1) return toList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}\n\n/**\n * Sorts elements in the list in-place according to
natural sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that

```

```

equal elements preserve their order relative to each other after sorting.\n */\npublic inline fun <T, R :
Comparable<R>> MutableList<T>.sortBy(crossinline selector: (T) -> R?): Unit {\n if (size > 1)
sortWith(compareBy(selector))\n}\n\n/**\n * Sorts elements in the list in-place descending according to natural sort
order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal
elements preserve their order relative to each other after sorting.\n */\npublic inline fun <T, R : Comparable<R>>
MutableList<T>.sortByDescending(crossinline selector: (T) -> R?): Unit {\n if (size > 1)
sortWith(compareByDescending(selector))\n}\n\n/**\n * Sorts elements in the list in-place descending according to
their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to
each other after sorting.\n */\npublic fun <T : Comparable<T>> MutableList<T>.sortDescending(): Unit {\n
sortWith(reverseOrder())\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort order.\n * \n *
The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\n
*\npublic fun <T : Comparable<T>> Iterable<T>.sorted(): List<T> {\n if (this is Collection) {\n if (size <= 1)
return this.toList()\n @Suppress("\nUNCHECKED_CAST")\n return (toTypedArray<Comparable<T>>())
as Array<T>).apply { sort() }.asList()\n }\n }\n return toMutableList().apply { sort() }\n}\n\n/**\n * Returns a list of
all elements sorted according to natural sort order of the value returned by specified [selector] function.\n * \n * The
sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\npublic
inline fun <T, R : Comparable<R>> Iterable<T>.sortedBy(crossinline selector: (T) -> R?): List<T> {\n return
sortedWith(compareBy(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending according to natural
sort order of the value returned by specified [selector] function.\n * \n * The sort is _stable_. It means that equal
elements preserve their order relative to each other after sorting.\n */\npublic inline fun <T, R : Comparable<R>>
Iterable<T>.sortedByDescending(crossinline selector: (T) -> R?): List<T> {\n return
sortedWith(compareByDescending(selector))\n}\n\n/**\n * Returns a list of all elements sorted descending
according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order
relative to each other after sorting.\n */\npublic fun <T : Comparable<T>> Iterable<T>.sortedDescending(): List<T>
{\n return sortedWith(reverseOrder())\n}\n\n/**\n * Returns a list of all elements sorted according to the specified
[comparator].\n * \n * The sort is _stable_. It means that equal elements preserve their order relative to each other
after sorting.\n */\npublic fun <T> Iterable<T>.sortedWith(comparator: Comparator<in T>): List<T> {\n if (this is
Collection) {\n if (size <= 1) return this.toList()\n @Suppress("\nUNCHECKED_CAST")\n return
(toTypedArray<Any?>()) as Array<T>).apply { sortWith(comparator) }.asList()\n }\n }\n return
toMutableList().apply { sortWith(comparator) }\n}\n\n/**\n * Returns an array of Boolean containing all of the
elements of this collection.\n */\npublic fun Collection<Boolean>.toBooleanArray(): BooleanArray {\n val result
= BooleanArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return
result\n}\n\n/**\n * Returns an array of Byte containing all of the elements of this collection.\n */\npublic fun
Collection<Byte>.toByteArray(): ByteArray {\n val result = ByteArray(size)\n var index = 0\n for (element in
this)\n result[index++] = element\n return result\n}\n\n/**\n * Returns an array of Char containing all of the
elements of this collection.\n */\npublic fun Collection<Char>.toCharArray(): CharArray {\n val result =
CharArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return
result\n}\n\n/**\n * Returns an array of Double containing all of the elements of this collection.\n */\npublic fun
Collection<Double>.toDoubleArray(): DoubleArray {\n val result = DoubleArray(size)\n var index = 0\n for
(element in this)\n result[index++] = element\n return result\n}\n\n/**\n * Returns an array of Float containing
all of the elements of this collection.\n */\npublic fun Collection<Float>.toFloatArray(): FloatArray {\n val result
= FloatArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return
result\n}\n\n/**\n * Returns an array of Int containing all of the elements of this collection.\n */\npublic fun
Collection<Int>.toIntArray(): IntArray {\n val result = IntArray(size)\n var index = 0\n for (element in this)\n
result[index++] = element\n return result\n}\n\n/**\n * Returns an array of Long containing all of the elements
of this collection.\n */\npublic fun Collection<Long>.toLongArray(): LongArray {\n val result =
LongArray(size)\n var index = 0\n for (element in this)\n result[index++] = element\n return
result\n}\n\n/**\n * Returns an array of Short containing all of the elements of this collection.\n */\npublic fun

```

```

Collection<Short>.toShortArray(): ShortArray {
    val result = ShortArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/** Returns a [Map] containing key-value pairs provided by [transform] function applied to elements of the given collection.
 * If any of two pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original collection.
 */
public inline fun <T, K, V> Iterable<T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/** Returns a [Map] containing the elements from the given collection indexed by the key returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original collection.
 */
public inline fun <T, K> Iterable<T>.associateBy(keySelector: (T) -> K): Map<K, T> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)
}

/** Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given collection.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original collection.
 */
public inline fun <T, K, V> Iterable<T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

/** Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function applied to each element of the given collection and value is the element itself.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 */
public inline fun <T, K, M : MutableMap<in K, in T>> Iterable<T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/** Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given collection.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 */
public inline fun <T, K, V, M : MutableMap<in K, in V>> Iterable<T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/** Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given collection.
 * If any of two pairs would have the same key the last one gets added to the map.
 */
public inline fun <T, K, V, M : MutableMap<in K, in V>> Iterable<T>.associateTo(destination: M, transform: (T) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/** Returns a [Map] where keys are elements from the given collection and values are produced by the [valueSelector] function applied to each element.
 * If any two elements are equal, the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original collection.
 */
@sample
samples.collections.Collections.Transformations.associateWith
@SinceKotlin("1.3")
public inline fun <K, V> Iterable<K>.associateWith(valueSelector: (K) -> V): Map<K, V> {
    val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/** Populates and returns the [destination] mutable map with key-value pairs for each element of the given collection, where key is the element itself and value is provided by the [valueSelector] function applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 */
@SinceKotlin("1.3")
public inline fun <K, V, M : MutableMap<in K, in V>> Iterable<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/** Appends all elements to the given [destination] collection.
 */
public fun <T, C : MutableCollection<in T>> Iterable<T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

```

```

destination\n}\n\n/**\n * Returns a [HashSet] of all elements.\n */\npublic fun <T> Iterable<T>.toHashSet():
HashSet<T> {\n    return toCollection(HashSet<T>(mapCapacity(collectionSizeOrDefault(12))))\n}\n\n/**\n *
Returns a [List] containing all elements.\n */\npublic fun <T> Iterable<T>.toList(): List<T> {\n    if (this is
Collection) {\n        return when (size) {\n            0 -> emptyList()\n            1 -> listOf(if (this is List) get(0) else
iterator().next())\n            else -> this.toMutableList()\n        }\n    }\n    return
this.toMutableList().optimizeReadOnlyList()\n}\n\n/**\n * Returns a [MutableList] filled with all elements of this
collection.\n */\npublic fun <T> Iterable<T>.toMutableList(): MutableList<T> {\n    if (this is Collection<T>)\n    return this.toMutableList()\n    return toCollection(ArrayList<T>())\n}\n\n/**\n * Returns a [MutableList] filled
with all elements of this collection.\n */\npublic fun <T> Collection<T>.toMutableList(): MutableList<T> {\n
return ArrayList(this)\n}\n\n/**\n * Returns a [Set] of all elements.\n */\n * The returned set preserves the element
iteration order of the original collection.\n */\npublic fun <T> Iterable<T>.toSet(): Set<T> {\n    if (this is
Collection) {\n        return when (size) {\n            0 -> emptySet()\n            1 -> setOf(if (this is List) this[0] else
iterator().next())\n            else -> toCollection(LinkedHashSet<T>(mapCapacity(size)))\n        }\n    }\n    return
toCollection(LinkedHashSet<T>()).optimizeReadOnlySet()\n}\n\n/**\n * Returns a single list of all elements
yielded from results of [transform] function being invoked on each element of original collection.\n */\npublic inline
fun <T, R> Iterable<T>.flatMap(transform: (T) -> Iterable<R>): List<R> {\n    return flatMapTo(ArrayList<R>(),
transform)\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each
element of original collection, to the given [destination].\n */\npublic inline fun <T, R, C : MutableCollection<in
R>> Iterable<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {\n    for (element in this) {\n        val
list = transform(element)\n        destination.addAll(list)\n    }\n    return destination\n}\n\n/**\n * Groups elements of
the original collection by the key returned by the given [keySelector] function\n */ applied to each element and
returns a map where each group key is associated with a list of corresponding elements.\n */\n * The returned map
preserves the entry iteration order of the keys produced from the original collection.\n */\n * @sample
samples.collections.Collections.Transformations.groupBy\n */\npublic inline fun <T, K>
Iterable<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {\n    return groupByTo(LinkedHashMap<K,
MutableList<T>>(), keySelector)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to
each element of the original collection\n */ by the key returned by the given [keySelector] function applied to the
element\n */ and returns a map where each group key is associated with a list of corresponding values.\n */\n * The
returned map preserves the entry iteration order of the keys produced from the original collection.\n */\n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <T, K, V>
Iterable<T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {\n    return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups elements
of the original collection by the key returned by the given [keySelector] function\n */ applied to each element and
puts to the [destination] map each group key associated with a list of corresponding elements.\n */\n * @return The
[destination] map.\n */\n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic inline
fun <T, K, M : MutableMap<in K, MutableList<T>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) ->
K): M {\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {
ArrayList<T>() }\n        list.add(element)\n    }\n    return destination\n}\n\n/**\n * Groups values returned by the
[valueTransform] function applied to each element of the original collection\n */ by the key returned by the given
[keySelector] function applied to the element\n */ and puts to the [destination] map each group key associated with a
list of corresponding values.\n */\n * @return The [destination] map.\n */\n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <T, K, V, M :
MutableMap<in K, MutableList<V>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K,
valueTransform: (T) -> V): M {\n    for (element in this) {\n        val key = keySelector(element)\n        val list =
destination.getOrPut(key) { ArrayList<V>() }\n        list.add(valueTransform(element))\n    }\n    return
destination\n}\n\n/**\n * Creates a [Grouping] source from a collection to be used later with one of group-and-fold
operations\n */ using the specified [keySelector] function to extract a key from each element.\n */\n * @sample
samples.collections.Grouping.groupingByEachCount\n */\n * @SinceKotlin("1.1")\n */\npublic inline fun <T, K>

```

```

Iterable<T>.groupBy(crossinline keySelector: (T) -> K): Grouping<T, K> {\n  return object : Grouping<T, K>
{\n    override fun sourceIterator(): Iterator<T> = this@groupBy.iterator()\n    override fun keyOf(element:
T): K = keySelector(element)\n  }}\n\n/**\n * Returns a list containing the results of applying the given
[transform] function\n * to each element in the original collection.\n * \n * @sample
samples.collections.Collections.Transformations.map\n * \n\npublic inline fun <T, R> Iterable<T>.map(transform:
(T) -> R): List<R> {\n  return mapTo<ArrayList<R>>(collectionSizeOrDefault(10)), transform)\n}\n\n/**\n *
Returns a list containing the results of applying the given [transform] function\n * to each element and its index in
the original collection.\n * @param [transform] function that takes the index of an element and the element itself\n *
and returns the result of the transform applied to the element.\n * \n\npublic inline fun <T, R>
Iterable<T>.mapIndexed(transform: (index: Int, T) -> R): List<R> {\n  return
mapIndexedTo<ArrayList<R>>(collectionSizeOrDefault(10)), transform)\n}\n\n/**\n * Returns a list containing only
the non-null results of applying the given [transform] function\n * to each element and its index in the original
collection.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns
the result of the transform applied to the element.\n * \n\npublic inline fun <T, R : Any>
Iterable<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): List<R> {\n  return
mapIndexedNotNullTo<ArrayList<R>>(), transform)\n}\n\n/**\n * Applies the given [transform] function to each
element and its index in the original collection\n * and appends only the non-null results to the given [destination].\n
* @param [transform] function that takes the index of an element and the element itself\n * and returns the result of
the transform applied to the element.\n * \n\npublic inline fun <T, R : Any, C : MutableCollection<in R>>
Iterable<T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {\n  forEachIndexed {
index, element -> transform(index, element)?.let { destination.add(it) } }\n  return destination\n}\n\n/**\n *
Applies the given [transform] function to each element and its index in the original collection\n * and appends the
results to the given [destination].\n * @param [transform] function that takes the index of an element and the
element itself\n * and returns the result of the transform applied to the element.\n * \n\npublic inline fun <T, R, C :
MutableCollection<in R>> Iterable<T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {\n  var
index = 0\n  for (item in this)\n    destination.add(transform(checkIndexOverflow(index++), item))\n  return
destination\n}\n\n/**\n * Returns a list containing only the non-null results of applying the given [transform]
function\n * to each element in the original collection.\n * \n\npublic inline fun <T, R : Any>
Iterable<T>.mapNotNull(transform: (T) -> R?): List<R> {\n  return mapNotNullTo<ArrayList<R>>(),
transform)\n}\n\n/**\n * Applies the given [transform] function to each element in the original collection\n * and
appends only the non-null results to the given [destination].\n * \n\npublic inline fun <T, R : Any, C :
MutableCollection<in R>> Iterable<T>.mapNotNullTo(destination: C, transform: (T) -> R?): C {\n  forEach {
element -> transform(element)?.let { destination.add(it) } }\n  return destination\n}\n\n/**\n * Applies the given
[transform] function to each element of the original collection\n * and appends the results to the given
[destination].\n * \n\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.mapTo(destination: C,
transform: (T) -> R): C {\n  for (item in this)\n    destination.add(transform(item))\n  return
destination\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original collection\n * into an
[IndexValue] containing the index of that element and the element itself.\n * \n\npublic fun <T>
Iterable<T>.withIndex(): Iterable<IndexedValue<T>> {\n  return IndexingIterable { iterator() }\n}\n\n/**\n *
Returns a list containing only distinct elements from the given collection.\n * \n * The elements in the resulting list
are in the same order as they were in the source collection.\n * \n\npublic fun <T> Iterable<T>.distinct(): List<T> {\n
return this.toMutableSet().toList()\n}\n\n/**\n * Returns a list containing only elements from the given collection\n
* having distinct keys returned by the given [selector] function.\n * \n * The elements in the resulting list are in the
same order as they were in the source collection.\n * \n\npublic inline fun <T, K> Iterable<T>.distinctBy(selector: (T)
-> K): List<T> {\n  val set = HashSet<K>()\n  val list = ArrayList<T>()\n  for (e in this) {\n    val key =
selector(e)\n    if (set.add(key))\n      list.add(e)\n  }\n  return list\n}\n\n/**\n * Returns a set containing all
elements that are contained by both this set and the specified collection.\n * \n * The returned set preserves the
element iteration order of the original collection.\n * \n\npublic infix fun <T> Iterable<T>.intersect(other:

```



```

Iterable<T>): Set<T> {\n    val set = this.toMutableSet()\n    set.retainAll(other)\n    return set}\n\n/**\n * Returns a set containing all elements that are contained by this collection and not contained by the specified collection.\n * \n * The returned set preserves the element iteration order of the original collection.\n */\npublic infix fun <T>
Iterable<T>.subtract(other: Iterable<T>): Set<T> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set}\n\n/**\n * Returns a mutable set containing all distinct elements from the given collection.\n * \n * The returned set preserves the element iteration order of the original collection.\n */\npublic fun <T>
Iterable<T>.toMutableSet(): MutableSet<T> {\n    return when (this) {\n        is Collection<T> ->
LinkedHashSet(this)\n        else -> toCollection(LinkedHashSet<T>())\n    }\n}\n\n/**\n * Returns a set containing all distinct elements from both collections.\n * \n * The returned set preserves the element iteration order of the original collection.\n * Those elements of the [other] collection that are unique are iterated in the end\n * in the order of the [other] collection.\n */\npublic infix fun <T> Iterable<T>.union(other: Iterable<T>): Set<T> {\n    val set = this.toMutableSet()\n    set.addAll(other)\n    return set}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\npublic inline fun <T>
Iterable<T>.all(predicate: (T) -> Boolean): Boolean {\n    if (this is Collection && isEmpty()) return true\n    for (element in this) if (!predicate(element)) return false\n    return true}\n\n/**\n * Returns `true` if collection has at least one element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n */\npublic fun <T>
Iterable<T>.any(): Boolean {\n    if (this is Collection) return !isEmpty()\n    return iterator().hasNext()\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\npublic inline fun <T>
Iterable<T>.any(predicate: (T) -> Boolean): Boolean {\n    if (this is Collection && isEmpty()) return false\n    for (element in this) if (predicate(element)) return true\n    return false}\n\n/**\n * Returns the number of elements in this collection.\n */\npublic fun <T> Iterable<T>.count(): Int {\n    if (this is Collection) return size\n    var count = 0\n    for (element in this) checkCountOverflow(++count)\n    return count}\n\n/**\n * Returns the number of elements in this collection.\n * \n * @kotlin.internal.InlineOnly\n */\npublic inline fun <T> Collection<T>.count(): Int {\n    return size}\n\n/**\n * Returns the number of elements matching the given [predicate].\n */\npublic inline fun <T>
Iterable<T>.count(predicate: (T) -> Boolean): Int {\n    if (this is Collection && isEmpty()) return 0\n    var count = 0\n    for (element in this) if (predicate(element)) checkCountOverflow(++count)\n    return count}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.\n */\npublic inline fun <T, R> Iterable<T>.fold(initial: R, operation: (acc: R, T) -> R): R {\n    var accumulator = initial\n    for (element in this) accumulator = operation(accumulator, element)\n    return accumulator}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original collection.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */\npublic inline fun <T, R> Iterable<T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {\n    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(checkIndexOverflow(index++), accumulator, element)\n    return accumulator}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.\n */\npublic inline fun <T, R> List<T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {\n    var accumulator = initial\n    if (!isEmpty()) {\n        val iterator = listIterator(size)\n        while (iterator.hasPrevious()) {\n            accumulator = operation(iterator.previous(), accumulator)\n        }\n    }\n    return accumulator}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element with its index in the original list and current accumulator value.\n * \n * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.\n */\npublic inline fun <T, R> List<T>.foldRightIndexed(initial: R, operation: (index: Int, T, acc: R) -> R): R {\n    var accumulator = initial\n    if (!isEmpty()) {\n        val iterator = listIterator(size)\n        while (iterator.hasPrevious()) {\n            val index = iterator.previousIndex()\n            accumulator = operation(index, iterator.previous(), accumulator)\n        }\n    }\n    return accumulator}\n\n/**\n * Performs the given [action] on each element.\n * \n * @kotlin.internal.HidesMembers\n */\npublic inline fun <T>

```

```

Iterable<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}
/**
 * Performs the given [action] on each element, providing sequential index with the element.
 * @param [action] function that takes the index of an element and the element itself
 * and performs the desired action on the element.
 */
public inline fun <T> Iterable<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) action(checkIndexOverflow(index++), item)
}
/**
 * Returns the largest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.1")
public fun Iterable<Double>.max(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    if (max.isNaN()) return max
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e.isNaN()) return e
        if (max < e) max = e
    }
    return max
}
/**
 * Returns the largest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.1")
public fun Iterable<Float>.max(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    if (max.isNaN()) return max
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e.isNaN()) return e
        if (max < e) max = e
    }
    return max
}
/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 * @sample samples.collections.Collections.Aggregates.maxBy
 */
public inline fun <T, R : Comparable<R>> Iterable<T>.maxBy(selector: (T) -> R): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxElem = iterator.next()
    if (!iterator.hasNext()) return maxElem
    var maxVal = selector(maxElem)
    do {
        val e = iterator.next()
        val v = selector(e)
        if (maxVal < v) {
            maxElem = e
            maxVal = v
        }
    } while (iterator.hasNext())
    return maxElem
}
/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun <T> Iterable<T>.maxWith(comparator: Comparator<in T>): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}
/**
 * Returns the smallest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.1")
public fun Iterable<Double>.min(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    if (min.isNaN()) return min
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e.isNaN()) return e
        if (min > e) min = e
    }
    return min
}
/**
 * Returns the smallest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.1")
public fun Iterable<Float>.min(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    if (min.isNaN()) return min
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e.isNaN()) return e
        if (min > e) min = e
    }
    return min
}
/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minBy
 */
public inline fun <T, R : Comparable<R>> Iterable<T>.minBy(selector: (T) -> R): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minElem = iterator.next()
    if (!iterator.hasNext()) return minElem
    var minVal = selector(minElem)
    do {
        val e = iterator.next()
        val v = selector(e)
        if (minVal > v) {
            minElem = e
            minVal = v
        }
    } while (iterator.hasNext())
    return minElem
}
/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun <T> Iterable<T>.minWith(comparator: Comparator<in T>): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
/**
 * Returns `true` if the collection has no elements.
 * @sample
 */

```

```

samples.collections.Collections.Aggregates.none\n *\npublic fun <T> Iterable<T>.none(): Boolean {\n  if (this is
Collection) return isEmpty()\n  return !iterator().hasNext()\n}\n\n/**\n * Returns `true` if no elements match the
given [predicate].\n *\n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n *\npublic
inline fun <T> Iterable<T>.none(predicate: (T) -> Boolean): Boolean {\n  if (this is Collection && isEmpty())
return true\n  for (element in this) if (predicate(element)) return false\n  return true\n}\n\n/**\n * Performs the
given [action] on each element and returns the collection itself afterwards.\n *\n@SinceKotlin("1.1")\npublic
inline fun <T, C : Iterable<T>> C.onEach(action: (T) -> Unit): C {\n  return apply { for (element in this)
action(element) }\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left
to right to current accumulator value and each element.\n *\npublic inline fun <S, T : S>
Iterable<T>.reduce(operation: (acc: S, T) -> S): S {\n  val iterator = this.iterator()\n  if (!iterator.hasNext()) throw
UnsupportedOperationException("Empty collection can't be reduced.")\n  var accumulator: S = iterator.next()\n  while (iterator.hasNext()) {\n    accumulator = operation(accumulator, iterator.next())\n  }\n  return
accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original collection.\n *\n@param
[operation] function that takes the index of an element, current accumulator value\n * and the element itself and
calculates the next accumulator value.\n *\npublic inline fun <S, T : S> Iterable<T>.reduceIndexed(operation:
(index: Int, acc: S, T) -> S): S {\n  val iterator = this.iterator()\n  if (!iterator.hasNext()) throw
UnsupportedOperationException("Empty collection can't be reduced.")\n  var index = 1\n  var accumulator: S =
iterator.next()\n  while (iterator.hasNext()) {\n    accumulator = operation(checkIndexOverflow(index++),
accumulator, iterator.next())\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with last element
and applying [operation] from right to left to each element and current accumulator value.\n *\npublic inline fun
<S, T : S> List<T>.reduceRight(operation: (T, acc: S) -> S): S {\n  val iterator = listIterator(size)\n  if
(!iterator.hasPrevious())\n    throw UnsupportedOperationException("Empty list can't be reduced.")\n  var
accumulator: S = iterator.previous()\n  while (iterator.hasPrevious()) {\n    accumulator =
operation(iterator.previous(), accumulator)\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting
with last element and applying [operation] from right to left\n * to each element with its index in the original list and
current accumulator value.\n *\n@param [operation] function that takes the index of an element, the element itself\n *
and current accumulator value, and calculates the next accumulator value.\n *\npublic inline fun <S, T : S>
List<T>.reduceRightIndexed(operation: (index: Int, T, acc: S) -> S): S {\n  val iterator = listIterator(size)\n  if
(!iterator.hasPrevious())\n    throw UnsupportedOperationException("Empty list can't be reduced.")\n  var
accumulator: S = iterator.previous()\n  while (iterator.hasPrevious()) {\n    val index = iterator.previousIndex()\n
accumulator = operation(index, iterator.previous(), accumulator)\n  }\n  return accumulator\n}\n\n/**\n *
Returns the sum of all values produced by [selector] function applied to each element in the collection.\n *\npublic
inline fun <T> Iterable<T>.sumBy(selector: (T) -> Int): Int {\n  var sum: Int = 0\n  for (element in this) {\n
sum += selector(element)\n  }\n  return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector]
function applied to each element in the collection.\n *\npublic inline fun <T> Iterable<T>.sumByDouble(selector:
(T) -> Double): Double {\n  var sum: Double = 0.0\n  for (element in this) {\n    sum += selector(element)\n
}\n  return sum\n}\n\n/**\n * Returns an original collection containing all the non-`null` elements, throwing an
[IllegalArgumentException] if there are any `null` elements.\n *\npublic fun <T : Any>
Iterable<T?>.requireNonNulls(): Iterable<T> {\n  for (element in this) {\n    if (element == null) {\n      throw
IllegalArgumentException("null element found in $this.")\n    }\n  }\n  @Suppress("UNCHECKED_CAST")\n  return this as Iterable<T>\n}\n\n/**\n * Returns an original collection
containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.\n *\npublic fun <T : Any> List<T?>.requireNonNulls(): List<T> {\n  for (element in this) {\n    if (element ==
null) {\n      throw IllegalArgumentException("null element found in $this.")\n    }\n  }\n  @Suppress("UNCHECKED_CAST")\n  return this as List<T>\n}\n\n/**\n * Splits this collection into a list of
lists each not exceeding the given [size].\n *\n * The last list in the resulting list may have less elements than the
given [size].\n *\n * @param size the number of elements to take in each list, must be positive and can be greater

```

than the number of elements in this collection.

```

\n * \n * @sample
samples.collections.Collections.Transformations.chunked
\n * \n * @SinceKotlin("1.2")\npublic fun <T>
Iterable<T>.chunked(size: Int): List<List<T>> {
\n    return windowed(size, size, partialWindows = true)\n}\n\n/**
\n * Splits this collection into several lists each not exceeding the given [size]
\n * and applies the given [transform] function to an each.
\n * \n * @return list of results of the [transform] applied to an each list.
\n * \n * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.
\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.
\n * The last list may have less elements than the given [size].
\n * \n * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.
\n * \n * @sample samples.text.Strings.chunkedTransform
\n * \n * @SinceKotlin("1.2")\npublic fun <T, R> Iterable<T>.chunked(size: Int, transform: (List<T>) -> R): List<R>
{
\n    return windowed(size, size, partialWindows = true, transform = transform)\n}\n\n/**
\n * Returns a list containing all elements of the original collection without the first occurrence of the given [element].
\n */\npublic operator fun <T> Iterable<T>.minus(element: T): List<T> {
\n    val result = ArrayList<T>(collectionSizeOrDefault(10))
\n    var removed = false
\n    return this.filterTo(result) { if (!removed && it == element) { removed = true; false } else true }\n}\n\n/**
\n * Returns a list containing all elements of the original collection except the elements contained in the given [elements] array.
\n * \n * The [elements] array may be converted to a [HashSet] to speed up the operation, thus the elements are required to have
\n * a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
\n */\npublic operator fun <T> Iterable<T>.minus(elements: Array<out T>): List<T> {
\n    if (elements.isEmpty()) return this.toList()\n    val other = elements.toHashSet()\n    return this.filterNot { it in other }\n}\n\n/**
\n * Returns a list containing all elements of the original collection except the elements contained in the given [elements] collection.
\n * \n * The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have
\n * a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
\n */\npublic operator fun <T> Iterable<T>.minus(elements: Iterable<T>): List<T> {
\n    val other = elements.convertToSetForSetOperationWith(this)\n    if (other.isEmpty())\n        return this.toList()\n    return this.filterNot { it in other }\n}\n\n/**
\n * Returns a list containing all elements of the original collection except the elements contained in the given [elements] sequence.
\n * \n * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have
\n * a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
\n */\npublic operator fun <T> Iterable<T>.minus(elements: Sequence<T>): List<T> {
\n    val other = elements.toHashSet()\n    if (other.isEmpty())\n        return this.toList()\n    return this.filterNot { it in other }\n}\n\n/**
\n * Returns a list containing all elements of the original collection without the first occurrence of the given [element].
\n */
\n * \n * @kotl.in.internal.InlineOnly\npublic inline fun <T> Iterable<T>.minusElement(element: T): List<T> {
\n    return minus(element)\n}\n\n/**
\n * Splits the original collection into pair of lists,
\n * where *first* list contains elements for which [predicate] yielded `true`,
\n * while *second* list contains elements for which [predicate] yielded `false`.
\n */\npublic inline fun <T> Iterable<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {
\n    val first = ArrayList<T>()\n    val second = ArrayList<T>()\n    for (element in this) {
\n        if (predicate(element))\n            first.add(element)\n        else\n            second.add(element)\n    }\n}\n\n/**
\n * Returns a list containing all elements of the original collection and then the given [element].
\n */\npublic operator fun <T> Iterable<T>.plus(element: T): List<T> {
\n    if (this is Collection)\n        return this.plus(element)\n    val result = ArrayList<T>()\n    result.addAll(this)\n    result.add(element)\n    return result\n}\n\n/**
\n * Returns a list containing all elements of the original collection and then the given [element].
\n */\npublic operator fun <T> Collection<T>.plus(element: T): List<T> {
\n    val result = ArrayList<T>(size + 1)\n    result.addAll(this)\n    result.add(element)\n    return result\n}\n\n/**
\n * Returns a list containing all elements of the original collection and then all elements of the given [elements] array.
\n */\npublic operator fun <T> Iterable<T>.plus(elements: Array<out T>): List<T> {
\n    if (this is Collection)\n        return this.plus(elements)\n    val result = ArrayList<T>()\n    result.addAll(this)\n    result.addAll(elements)\n    return result\n}\n\n/**
\n * Returns a list containing all elements of the original collection and then all elements of the given [elements] array.
\n */\npublic

```

```

operator fun <T> Collection<T>.plus(elements: Array<out T>): List<T> {\n    val result = ArrayList<T>(this.size +
elements.size)\n    result.addAll(this)\n    result.addAll(elements)\n    return result}\n\n/**\n * Returns a list
containing all elements of the original collection and then all elements of the given [elements] collection.\n
*/\npublic operator fun <T> Iterable<T>.plus(elements: Iterable<T>): List<T> {\n    if (this is Collection) return
this.plus(elements)\n    val result = ArrayList<T>()\n    result.addAll(this)\n    result.addAll(elements)\n    return
result}\n\n/**\n * Returns a list containing all elements of the original collection and then all elements of the given
[elements] collection.\n
*/\npublic operator fun <T> Collection<T>.plus(elements: Iterable<T>): List<T> {\n    if
(elements is Collection) {\n        val result = ArrayList<T>(this.size + elements.size)\n        result.addAll(this)\n
result.addAll(elements)\n        return result\n    } else {\n        val result = ArrayList<T>(this)\n
result.addAll(elements)\n        return result\n    }\n}\n\n/**\n * Returns a list containing all elements of the original
collection and then all elements of the given [elements] sequence.\n
*/\npublic operator fun <T>
Iterable<T>.plus(elements: Sequence<T>): List<T> {\n    val result = ArrayList<T>()\n    result.addAll(this)\n
result.addAll(elements)\n    return result}\n\n/**\n * Returns a list containing all elements of the original
collection and then all elements of the given [elements] sequence.\n
*/\npublic operator fun <T>
Collection<T>.plus(elements: Sequence<T>): List<T> {\n    val result = ArrayList<T>(this.size + 10)\n
result.addAll(this)\n    result.addAll(elements)\n    return result}\n\n/**\n * Returns a list containing all elements
of the original collection and then the given [element].\n
*/\n@kotlin.internal.InlineOnly\npublic inline fun <T>
Iterable<T>.plusElement(element: T): List<T> {\n    return plus(element)}\n\n/**\n * Returns a list containing all
elements of the original collection and then the given [element].\n
*/\n@kotlin.internal.InlineOnly\npublic inline fun
<T> Collection<T>.plusElement(element: T): List<T> {\n    return plus(element)}\n\n/**\n * Returns a list of
snapshots of the window of the given [size]\n * sliding along this collection with the given [step], where each\n *
snapshot is a list.\n * \n * Several last lists may have less elements than the given [size].\n * \n * Both [size] and
[step] must be positive and can be greater than the number of elements in this collection.\n * @param size the
number of elements to take in each window\n * @param step the number of elements to move the window forward
by on an each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the
end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample
samples.collections.Sequences.Transformations.takeWindows\n
*/\n@SinceKotlin("1.2")\npublic fun <T>
Iterable<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): List<List<T>> {\n
checkWindowSizeStep(size, step)\n    if (this is RandomAccess && this is List) {\n        val thisSize = this.size\n
val result = ArrayList<List<T>>((thisSize + step - 1) / step)\n        var index = 0\n        while (index < thisSize) {\n
            val windowSize = size.coerceAtMost(thisSize - index)\n            if (windowSize < size && !partialWindows)
break\n            result.add(List(windowSize) { this[it + index] })\n            index += step\n        }\n        return result\n
    }\n    val result = ArrayList<List<T>>()\n    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer =
false).forEach {\n        result.add(it)\n    }\n    return result}\n\n/**\n * Returns a list of results of applying the
given [transform] function to\n * an each list representing a view over the window of the given [size]\n * sliding
along this collection with the given [step].\n * \n * Note that the list passed to the [transform] function is ephemeral
and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made
a snapshot of it.\n * Several last lists may have less elements than the given [size].\n * \n * Both [size] and [step]
must be positive and can be greater than the number of elements in this collection.\n * @param size the number of
elements to take in each window\n * @param step the number of elements to move the window forward by on an
each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if
any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample
samples.collections.Sequences.Transformations.averageWindows\n
*/\n@SinceKotlin("1.2")\npublic fun <T, R>
Iterable<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (List<T>) -> R):
List<R> {\n    checkWindowSizeStep(size, step)\n    if (this is RandomAccess && this is List) {\n        val thisSize =
this.size\n        val result = ArrayList<R>((thisSize + step - 1) / step)\n        val window = MovingSubList(this)\n
var index = 0\n        while (index < thisSize) {\n            window.move(index, (index +
size).coerceAtMost(thisSize))\n            if (!partialWindows && window.size < size) break\n

```

```

result.add(transform(window))\n      index += step\n    }\n    return result\n  }\n  val result =
ArrayList<R>()\n  windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = true).forEach { \n
result.add(transform(it))\n  }\n  return result\n}\n\n/**\n * Returns a list of pairs built from the elements of `this`
collection and the [other] array with the same index.\n * The returned list has length of the shortest collection.\n * \n
*\n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <T, R> Iterable<T>.zip(other:
Array<out R>): List<Pair<T, R>> { \n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values
built from the elements of `this` collection and the [other] array with the same index\n * using the provided
[transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n
*\n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <T, R, V>
Iterable<T>.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> { \n  val arraySize = other.size\n  val
list = ArrayList<V>(minOf(collectionSizeOrDefault(10), arraySize))\n  var i = 0\n  for (element in this) { \n    if
(i >= arraySize) break\n    list.add(transform(element, other[i++]))\n  }\n  return list\n}\n\n/**\n * Returns a list
of pairs built from the elements of `this` collection and [other] collection with the same index.\n * The returned list
has length of the shortest collection.\n * \n
*\n * @sample samples.collections.Iterables.Operations.zipIterable\n */\npublic infix fun <T, R> Iterable<T>.zip(other: Iterable<R>): List<Pair<T, R>> { \n  return zip(other) { t1, t2 ->
t1 to t2 }\n}\n\n/**\n * Returns a list of values built from the elements of `this` collection and the [other] collection
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest collection.\n * \n
*\n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n */\npublic inline fun <T, R, V>
Iterable<T>.zip(other: Iterable<R>, transform: (a: T, b: R) -> V): List<V> { \n  val first = iterator()\n  val second
= other.iterator()\n  val list = ArrayList<V>(minOf(collectionSizeOrDefault(10),
other.collectionSizeOrDefault(10)))\n  while (first.hasNext() && second.hasNext()) { \n
list.add(transform(first.next(), second.next()))\n  }\n  return list\n}\n\n/**\n * Returns a list of pairs of each two
adjacent elements in this collection.\n * \n
*\n * The returned list is empty if this collection contains less than two
elements.\n * \n
*\n * @sample samples.collections.Collections.Transformations.zipWithNext\n */\npublic fun <T> Iterable<T>.zipWithNext(): List<Pair<T, T>> { \n  return zipWithNext
{ a, b -> a to b }\n}\n\n/**\n * Returns a list containing the results of applying the given [transform] function\n * to
an each pair of two adjacent elements in this collection.\n * \n
*\n * The returned list is empty if this collection contains
less than two elements.\n * \n
*\n * @sample
samples.collections.Collections.Transformations.zipWithNextToFindDeltas\n */\npublic inline fun <T, R> Iterable<T>.zipWithNext(transform: (a: T, b: T) -> R): List<R> { \n  val iterator = iterator()\n  if
(!iterator.hasNext()) return emptyList()\n  val result = mutableListOf<R>()\n  var current = iterator.next()\n  while (iterator.hasNext()) { \n    val next = iterator.next()\n    result.add(transform(current, next))\n    current
= next\n  }\n  return result\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and
using the given [prefix] and [postfix] if supplied.\n * \n
*\n * If the collection could be huge, you can specify a non-
negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the
[truncated] string (which defaults to "...").\n * \n
*\n * @sample
samples.collections.Collections.Transformations.joinTo\n */\npublic fun <T, A : Appendable>
Iterable<T>.joinTo(buffer: A, separator: CharSequence = "\n", prefix: CharSequence = "", postfix: CharSequence
= "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): A { \n
buffer.append(prefix)\n  var count = 0\n  for (element in this) { \n    if (++count > 1) buffer.append(separator)\n
    if (limit < 0 || count <= limit) { \n      buffer.appendElement(element, transform)\n    } else break\n  }\n  if
(limit >= 0 && count > limit) buffer.append(truncated)\n  buffer.append(postfix)\n  return buffer\n}\n\n/**\n *
Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if
supplied.\n * \n
*\n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only
the first [limit]\n * elements will be appended, followed by the [truncated] string (which defaults to "...").\n * \n
*\n * @sample samples.collections.Collections.Transformations.joinToString\n */\npublic fun <T>
Iterable<T>.joinToString(separator: CharSequence = "\n", prefix: CharSequence = "", postfix: CharSequence =

```

```

 "\", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): String {
    return
    joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/** Returns this collection as an [Iterable]. */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.asIterable(): Iterable<T> {
    return this
}

/** Creates a [Sequence] instance that wraps the original collection returning its elements when being iterated. */
@sample
samples.collections.Sequences.Building.sequenceFromCollection
public fun <T> Iterable<T>.asSequence(): Sequence<T> {
    return Sequence { this.iterator() }
}

/** Returns an average value of elements in the collection. */
@kotlin.jvm.JvmName("averageOfByte")
public fun Iterable<Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/** Returns an average value of elements in the collection. */
@kotlin.jvm.JvmName("averageOfShort")
public fun Iterable<Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/** Returns an average value of elements in the collection. */
@kotlin.jvm.JvmName("averageOfInt")
public fun Iterable<Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/** Returns an average value of elements in the collection. */
@kotlin.jvm.JvmName("averageOfLong")
public fun Iterable<Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/** Returns an average value of elements in the collection. */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Iterable<Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/** Returns an average value of elements in the collection. */
@kotlin.jvm.JvmName("averageOfDouble")
public fun Iterable<Double>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/** Returns the sum of all elements in the collection. */
@kotlin.jvm.JvmName("sumOfByte")
public fun Iterable<Byte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/** Returns the sum of all elements in the collection. */
@kotlin.jvm.JvmName("sumOfShort")
public fun Iterable<Short>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/** Returns the sum of all elements in the collection. */
@kotlin.jvm.JvmName("sumOfInt")
public fun Iterable<Int>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/** Returns the sum of all elements in the collection. */
@kotlin.jvm.JvmName("sumOfLong")
public fun Iterable<Long>.sum(): Long {
    var sum: Long = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/** Returns the sum of all elements in the collection. */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Iterable<Float>.sum(): Float {
    var sum: Float = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/** Returns the sum of all elements in the collection. */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Iterable<Double>.sum(): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package kotlin.collections
import kotlin.comparisons.naturalOrder
import kotlin.random.Random

/** Returns the array if it's not `null`, or an empty array otherwise. */
@kotlin.internal.InlineOnly
public actual inline fun <T> Array<out T>?.orEmpty(): Array<out T> = this ?: emptyArray<T>()

@kotlin.internal.InlineOnly
public actual inline fun <T> Collection<T>.toArray(): Array<T> = copyToArray(this)

@JsName("copyToArray")
@PublishedApi
internal fun <T>

```

```

copyToArray(collection: Collection<T>): Array<T> {
    return if (collection.asDynamic().toArray !== undefined)
        collection.asDynamic().toArray().unsafeCast<Array<T>>>()
    else
        copyToArrayImpl(collection).unsafeCast<Array<T>>>()
}

@JsName("copyToArrayImpl")
internal actual fun copyToArrayImpl(collection: Collection<*>): Array<Any?> {
    val array = emptyArray<Any?>>()
    val iterator = collection.iterator()
    while (iterator.hasNext())
        array.asDynamic().push(iterator.next())
    return array
}

@JsName("copyToExistingArrayImpl")
internal actual fun <T>
copyToArrayImpl(collection: Collection<*>, array: Array<T>): Array<T> {
    if (array.size < collection.size)
        return copyToArrayImpl(collection).unsafeCast<Array<T>>>()
    val iterator = collection.iterator()
    var index = 0
    while (iterator.hasNext()) {
        array[index++] = iterator.next().unsafeCast<T>>()
    }
    if (index < array.size) {
        array[index] = null.unsafeCast<T>>()
    }
    return array
}

/**
 * Returns an immutable list containing only the specified object [element].
 */
public fun <T> listOf(element: T): List<T> = arrayListOf(element)

/**
 * Returns an immutable set containing only the specified object [element].
 */
public fun <T> setOf(element: T): Set<T> = hashSetOf(element)

/**
 * Returns an immutable map, mapping only the specified key to the specified value.
 */
public fun <K, V> mapOf(pair: Pair<K, V>): Map<K, V> = hashMapOf(pair)

/**
 * Fills the list with the provided [value].
 * Each element in the list gets replaced with the [value].
 */
@SinceKotlin("1.2")
public actual fun <T> MutableList<T>.fill(value: T): Unit {
    for (index in 0..lastIndex)
        this[index] = value
}

/**
 * Randomly shuffles elements in this list.
 * See:
 * https://en.wikipedia.org/wiki/Fisher%20%80%93Yates_shuffle#The_modern_algorithm
 */
@SinceKotlin("1.2")
public actual fun <T> MutableList<T>.shuffle(): Unit =
    shuffle(Random)

/**
 * Returns a new list with the elements of this list randomly shuffled.
 */
@SinceKotlin("1.2")
public actual fun <T> Iterable<T>.shuffled(): List<T> = toMutableList().apply {
    shuffle()
}

/**
 * Sorts elements in the list in-place according to their natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public actual fun <T : Comparable<T>> MutableList<T>.sort(): Unit {
    collectionsSort(this, naturalOrder())
}

/**
 * Sorts elements in the list in-place according to the order specified with [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public actual fun <T> MutableList<T>.sortWith(comparator: Comparator<in T>): Unit {
    collectionsSort(this, comparator)
}

private fun <T> collectionsSort(list: MutableList<T>, comparator: Comparator<in T>) {
    if (list.size <= 1) return
    val array = copyToArray(list)
    sortArrayWith(array, comparator)
    for (i in 0 until array.size)
        list[i] = array[i]
}

internal actual fun <T> arrayOfNulls(reference: Array<T>, size: Int): Array<T> {
    return arrayOfNulls<Any>(size).unsafeCast<Array<T>>>()
}

@SinceKotlin("1.3")
@PublishedApi
@JsName("arrayCopy")
internal fun <T> arrayCopy(source: Array<out T>, destination: Array<in T>, destinationOffset: Int, startIndex: Int, endIndex: Int) {
    AbstractList.checkRangeIndexes(startIndex, endIndex, source.size)
    val rangeSize = endIndex - startIndex
    AbstractList.checkRangeIndexes(destinationOffset, destinationOffset + rangeSize, destination.size)
    if (js("ArrayBuffer").isView(destination) && js("ArrayBuffer").isView(source)) {
        val subrange = source.asDynamic().subarray(startIndex, endIndex)
        destination.asDynamic().set(subrange, destinationOffset)
    } else {
        if (source !== destination || destinationOffset <= startIndex) {
            for (index in 0 until rangeSize) {
                destination[destinationOffset + index] = source[startIndex + index]
            }
        } else {
            for (index in rangeSize - 1 downTo 0) {
                destination[destinationOffset + index] = source[startIndex + index]
            }
        }
    }
}

// no singleton map implementation in js, return map as
is@Suppress("NOTHING_TO_INLINE")
internal actual inline fun <K, V> Map<K, V>.toSingletonMapOrSelf(): Map<K, V> = this

@Suppress("NOTHING_TO_INLINE")
internal actual inline fun <K, V> Map<out K, V>.toSingletonMap(): Map<K, V> =
    this.toMutableMap()

@Suppress("NOTHING_TO_INLINE")
internal actual inline fun <T> Array<out T>.copyToArrayOfAny(isVarargs: Boolean): Array<out Any?> =
    if (isVarargs) // no need to

```



```
copy vararg array in JS
    this else
        this.copyOf()
@PublishedApi internal actual
fun checkIndexOverflow(index: Int): Int {
    if (index < 0)
        throwIndexOverflow()
}
return index
}
@PublishedApi internal actual
fun checkCountOverflow(count: Int): Int {
    if (count < 0)
        throwCountOverflow()
}
return count
}
"/**
 * Copyright 2010-2018 JetBrains
 s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0
 license that can be found in the license/LICENSE.txt file.
```

```
*/
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")
package
kotlin.collections
/**
 * Returns the given iterator itself. This allows to use an instance of iterator in a
 `for` loop.
 * @sample samples.collections.Iterators.iterator
 */
@kotlin.internal.InlineOnly public inline
operator fun <T> Iterator<T>.iterator(): Iterator<T> = this
 * Returns an [Iterator] that wraps each
 element produced by the original iterator
 * into an [IndexedValue] containing the index of that element and the
 element itself.
 */
@sample samples.collections.Iterators.withIndexIterator
public fun <T>
Iterator<T>.withIndex(): Iterator<IndexedValue<T>> = IndexingIterator(this)
 * Performs the given
 [operation] on each element of this [Iterator].
 */
@sample samples.collections.Iterators.forEachIterator
public inline fun <T> Iterator<T>.forEach(operation: (T) -> Unit): Unit {
    for (element in this)
        operation(element)
}
/**
 * Iterator transforming original `iterator` into iterator of [IndexedValue],
 counting index from zero.
 */
internal class IndexingIterator<out T>(private val iterator: Iterator<T>) :
Iterator<IndexedValue<T>> {
    private var index = 0
    final override fun hasNext(): Boolean =
        iterator.hasNext()
    final override fun next(): IndexedValue<T> = IndexedValue(checkIndexOverflow(index++),
        iterator.next())
}
"/**
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
 contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 license/LICENSE.txt file.
```

```
*/
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("ComparisonsKt")
package
kotlin.comparisons
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
import kotlin.random.
/**
 * Returns the
 greater of two values.
 * If values are equal, returns the first one.
 */
@SinceKotlin("1.1") public expect fun
<T : Comparable<T>> maxOf(a: T, b: T): T
/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Byte, b: Byte):
Byte
/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Short, b: Short): Short
/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Int, b: Int): Int
/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Long, b: Long): Long
/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Float, b: Float):
Float
/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Double, b: Double):
Double
/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1") public expect fun <T :
Comparable<T>> maxOf(a: T, b: T, c: T): T
/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Byte, b: Byte, c: Byte):
Byte
/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Short, b: Short, c:
Short): Short
/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Int, b: Int, c: Int):
Int
/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Long, b: Long, c: Long): Long
/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Float, b: Float, c:
Float): Float
/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1") @kotlin.internal.InlineOnly public expect inline fun maxOf(a: Double, b: Double, c:
```

```

Double): Double\n\n/**\n * Returns the greater of three values according to the order specified by the given
[comparator].\n */\n@SinceKotlin("1.1")\npublic fun <T> maxOf(a: T, b: T, c: T, comparator: Comparator<in T>):
T {\n    return maxOf(a, maxOf(b, c, comparator), comparator)\n}\n\n/**\n * Returns the greater of two values
according to the order specified by the given [comparator].\n * If values are equal, returns the first one.\n
*/\n@SinceKotlin("1.1")\npublic fun <T> maxOf(a: T, b: T, comparator: Comparator<in T>): T {\n    return if
(comparator.compare(a, b) >= 0) a else b\n}\n\n/**\n * Returns the smaller of two values.\n * If values are equal,
returns the first one.\n */\n@SinceKotlin("1.1")\npublic expect fun <T : Comparable<T>> minOf(a: T, b: T):
T\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic
expect inline fun minOf(a: Byte, b: Byte): Byte\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Short, b: Short):
Short\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Int, b: Int): Int\n\n/**\n
* Returns the smaller of two values.\n */\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic expect inline
fun minOf(a: Long, b: Long): Long\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Float, b: Float):
Float\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Double, b: Double):
Double\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin("1.1")\npublic expect fun <T :
Comparable<T>> minOf(a: T, b: T, c: T): T\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Byte, b: Byte, c: Byte):
Byte\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Short, b: Short, c:
Short): Short\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Int, b: Int, c: Int):
Int\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic
expect inline fun minOf(a: Long, b: Long, c: Long): Long\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Float, b: Float, c: Float):
Float\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Double, b: Double, c:
Double): Double\n\n/**\n * Returns the smaller of three values according to the order specified by the given
[comparator].\n */\n@SinceKotlin("1.1")\npublic fun <T> minOf(a: T, b: T, c: T, comparator: Comparator<in T>):
T {\n    return minOf(a, minOf(b, c, comparator), comparator)\n}\n\n/**\n * Returns the smaller of two values
according to the order specified by the given [comparator].\n * If values are equal, returns the first one.\n
*/\n@SinceKotlin("1.1")\npublic fun <T> minOf(a: T, b: T, comparator: Comparator<in T>): T {\n    return if
(comparator.compare(a, b) <= 0) a else b\n}\n\n"/**\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin
Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\n
*/\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("MapsKt")\n\npackage
kotlin.collections\n\n/\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n\nimport kotlin.random.*\nimport
kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns a [List] containing all key-value pairs.\n
*/\npublic fun <K, V> Map<out K, V>.toList(): List<Pair<K, V>> {\n    if (size == 0)\n        return emptyList()\n
    val iterator = entries.iterator()\n    if (!iterator.hasNext())\n        return emptyList()\n    val first = iterator.next()\n    if
(!iterator.hasNext())\n        return listOf(first.toPair())\n    val result = ArrayList<Pair<K, V>>(size)\n
    result.add(first.toPair())\n    do {\n        result.add(iterator.next().toPair())\n    } while (iterator.hasNext())\n    return
result\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being invoked
on each entry of original map.\n */\npublic inline fun <K, V, R> Map<out K, V>.flatMap(transform: (Map.Entry<K,
V>) -> Iterable<R>): List<R> {\n    return flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Appends all

```

elements yielded from results of [transform] function being invoked on each entry of original map, to the given [destination].

```

public inline fun <K, V, R, C : MutableCollection<in R>> Map<out K,
V>.flatMapTo(destination: C, transform: (Map.Entry<K, V>) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

```

* Returns a list containing the results of applying the given [transform] function * to each entry in the original map.

```

@sample samples.collections.Maps.Transformations.mapToList
public inline fun <K, V, R> Map<out K,
V>.map(transform: (Map.Entry<K, V>) -> R): List<R> {
    return mapTo(ArrayList<R>(size),
transform)
}

```

* Returns a list containing only the non-null results of applying the given [transform] function * to each entry in the original map.

```

public inline fun <K, V, R : Any> Map<out K,
V>.mapNotNull(transform: (Map.Entry<K, V>) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(),
transform)
}

```

* Applies the given [transform] function to each entry in the original map * and appends only the non-null results to the given [destination].

```

public inline fun <K, V, R : Any, C : MutableCollection<in
R>> Map<out K, V>.mapNotNullTo(destination: C, transform: (Map.Entry<K, V>) -> R?): C {
    for (element in this) {
        transform(element)?.let { destination.add(it) }
    }
    return destination
}

```

* Applies the given [transform] function to each entry of the original map * and appends the results to the given [destination].

```

public inline fun <K, V, R, C : MutableCollection<in R>> Map<out K, V>.mapTo(destination: C, transform:
(Map.Entry<K, V>) -> R): C {
    for (item in this) {
        destination.add(transform(item))
    }
    return destination
}

```

* Returns `true` if all entries match the given [predicate].

```

@sample samples.collections.Collections.Aggregates.all
public inline fun <K, V> Map<out K, V>.all(predicate:
(Map.Entry<K, V>) -> Boolean): Boolean {
    if (isEmpty()) return true
    for (element in this) if (!predicate(element)) return false
    return true
}

```

* Returns `true` if map has at least one entry.

```

@sample samples.collections.Collections.Aggregates.any
public fun <K, V> Map<out K, V>.any(): Boolean {
    return !isEmpty()
}

```

* Returns `true` if at least one entry matches the given [predicate].

```

@sample samples.collections.Collections.Aggregates.anyWithPredicate
public inline fun <K, V> Map<out K,
V>.any(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {
    if (isEmpty()) return false
    for (element in this) if (predicate(element)) return true
    return false
}

```

* Returns the number of entries in this map.

```

@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.count(): Int {
    return size
}

```

* Returns the number of entries matching the given [predicate].

```

public inline fun <K, V> Map<out K,
V>.count(predicate: (Map.Entry<K, V>) -> Boolean): Int {
    if (isEmpty()) return 0
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

```

* Performs the given [action] on each entry.

```

@kotlin.internal.HidesMembers
public inline fun <K, V> Map<out K, V>.forEach(action:
(Map.Entry<K, V>) -> Unit): Unit {
    for (element in this) action(element)
}

```

* Returns the first entry yielding the largest value of the given function or `null` if there are no entries.

```

@sample samples.collections.Collections.Aggregates.maxBy
@kotlin.internal.InlineOnly
public inline fun <K, V, R : Comparable<R>> Map<out K, V>.maxBy(selector: (Map.Entry<K, V>) -> R): Map.Entry<K, V>? {
    return entries.maxBy(selector)
}

```

* Returns the first entry having the largest value according to the provided [comparator] or `null` if there are no entries.

```

@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out
K, V>.maxWith(comparator: Comparator<in Map.Entry<K, V>>): Map.Entry<K, V>? {
    return entries.maxWith(comparator)
}

```

* Returns the first entry yielding the smallest value of the given function or `null` if there are no entries.

```

@sample samples.collections.Collections.Aggregates.minBy
public inline fun <K, V, R : Comparable<R>> Map<out K, V>.minBy(selector: (Map.Entry<K, V>) -> R): Map.Entry<K,
V>? {
    return entries.minBy(selector)
}

```

* Returns the first entry having the smallest value according to the provided [comparator] or `null` if there are no entries.

```

public fun <K, V> Map<out K,
V>.minWith(comparator: Comparator<in Map.Entry<K, V>>): Map.Entry<K, V>? {
    return entries.minWith(comparator)
}

```

* Returns `true` if the map has no entries.

```

@sample samples.collections.Collections.Aggregates.none
public fun <K, V> Map<out K, V>.none(): Boolean {
    return isEmpty()
}

```

* Returns `true` if no entries match the given [predicate].

```

@sample samples.collections.Collections.Aggregates.noneWithPredicate
public inline fun <K, V> Map<out K,

```

```

V>.none(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {\n  if (isEmpty()) return true\n  for (element in
this) if (predicate(element)) return false\n  return true}\n\n/**\n * Performs the given [action] on each entry and
returns the map itself afterwards.\n */\n@SinceKotlin("1.1")\npublic inline fun <K, V, M : Map<out K, V>>
M.onEach(action: (Map.Entry<K, V>) -> Unit): M {\n  return apply { for (element in this) action(element)
}\n}\n\n/**\n * Creates an [Iterable] instance that wraps the original map returning its entries when being iterated.\n
*/\n@kotlin.internal.InlineOnly\npublic inline fun <K, V> Map<out K, V>.asIterable(): Iterable<Map.Entry<K,
V>> {\n  return entries}\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original map returning its entries
when being iterated.\n */\npublic fun <K, V> Map<out K, V>.asSequence(): Sequence<Map.Entry<K, V>> {\n
return entries.asSequence()\n}\n}\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\npackage
kotlin.ranges\r\n\r\n@SinceKotlin("1.1")\r\n@kotlin.internal.InlineOnly\r\npublic inline operator fun
Float.rangeTo(that: Float): ClosedFloatingPointRange<Float> =\r\n
this.toDouble().rangeTo(that.toDouble()).unsafeCast<ClosedFloatingPointRange<Float>>()\r\n", "/*\n * Copyright
2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed
by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*/\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("SequencesKt")\n\npackage
kotlin.sequences\n\n/\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n\nimport kotlin.random.*\n\n/**\n * Returns
`true` if [element] is found in the sequence.\n */\n * The operation is _terminal_.\n */\npublic operator fun
<@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {\n  return indexOf(element)
>= 0\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the
[index] is out of bounds of this sequence.\n */\n * The operation is _terminal_.\n */\n * @sample
samples.collections.Collections.Elements.elementAt\n */\npublic fun <T> Sequence<T>.elementAt(index: Int): T
{\n  return elementAtOrElse(index) { throw IndexOutOfBoundsException("Sequence doesn't contain element at
index $index.") }\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue]
function if the [index] is out of bounds of this sequence.\n */\n * The operation is _terminal_.\n */\n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n */\npublic fun <T>
Sequence<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {\n  if (index < 0)\n    return
defaultValue(index)\n  val iterator = iterator()\n  var count = 0\n  while (iterator.hasNext()) {\n    val element
= iterator.next()\n    if (index == count++)\n      return element\n  }\n  return
defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of
this sequence.\n */\n * The operation is _terminal_.\n */\n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n */\npublic fun <T>
Sequence<T>.elementOrNull(index: Int): T? {\n  if (index < 0)\n    return null\n  val iterator = iterator()\n
var count = 0\n  while (iterator.hasNext()) {\n    val element = iterator.next()\n    if (index == count++)\n
return element\n  }\n  return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null`
if no such element was found.\n */\n * The operation is _terminal_.\n */\n@kotlin.internal.InlineOnly\npublic inline
fun <T> Sequence<T>.find(predicate: (T) -> Boolean): T? {\n  return firstOrNull(predicate)\n}\n\n/**\n * Returns
the last element matching the given [predicate], or `null` if no such element was found.\n */\n * The operation is
_terminal_.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.findLast(predicate: (T) ->
Boolean): T? {\n  return lastOrNull(predicate)\n}\n\n/**\n * Returns first element.\n */\n * @throws
[NoSuchElementException] if the sequence is empty.\n */\n * The operation is _terminal_.\n */\npublic fun <T>
Sequence<T>.first(): T {\n  val iterator = iterator()\n  if (!iterator.hasNext())\n    throw
NoSuchElementException("Sequence is empty.")\n  return iterator.next()\n}\n\n/**\n * Returns the first element
matching the given [predicate].\n */\n * @throws [NoSuchElementException] if no such element is found.\n */\n * The
operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.first(predicate: (T) -> Boolean): T {\n  for
(element in this) if (predicate(element)) return element\n  throw NoSuchElementException("Sequence contains no

```

element matching the predicate.)\n}\n\n/**\n * Returns the first element, or `null` if the sequence is empty.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.firstOrNull(): T? {\n val iterator = iterator()\n if (!iterator.hasNext())\n return null\n return iterator.next()\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element was not found.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.firstOrNull(predicate: (T) -> Boolean): T? {\n for (element in this) if (predicate(element))\n return element\n return null\n}\n\n/**\n * Returns first index of [element], or -1 if the sequence does not contain element.\n *\n * The operation is _terminal_.\n */\npublic fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOf(element: T): Int {\n var index = 0\n for (item in this) {\n checkIndexOverflow(index)\n if (element == item)\n return index\n index++\n }\n return -1\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the sequence does not contain such element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.indexOfFirst(predicate: (T) -> Boolean): Int {\n var index = 0\n for (item in this) {\n checkIndexOverflow(index)\n if (predicate(item))\n return index\n index++\n }\n return -1\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the sequence does not contain such element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.indexOfLast(predicate: (T) -> Boolean): Int {\n var lastIndex = -1\n var index = 0\n for (item in this) {\n checkIndexOverflow(index)\n if (predicate(item))\n lastIndex = index\n index++\n }\n return lastIndex\n}\n\n/**\n * Returns the last element.\n * @throws [NoSuchElementException] if the sequence is empty.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.last(): T {\n val iterator = iterator()\n if (!iterator.hasNext())\n throw NoSuchElementException("Sequence is empty.")\n var last = iterator.next()\n while (iterator.hasNext())\n last = iterator.next()\n return last\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.last(predicate: (T) -> Boolean): T {\n var last: T? = null\n var found = false\n for (element in this) {\n if (predicate(element))\n last = element\n found = true\n }\n if (!found) throw NoSuchElementException("Sequence contains no element matching the predicate.")\n @Suppress("UNCHECKED_CAST")\n return last as T\n}\n\n/**\n * Returns last index of [element], or -1 if the sequence does not contain element.\n *\n * The operation is _terminal_.\n */\npublic fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.lastIndexOf(element: T): Int {\n var lastIndex = -1\n var index = 0\n for (item in this) {\n checkIndexOverflow(index)\n if (element == item)\n lastIndex = index\n index++\n }\n return lastIndex\n}\n\n/**\n * Returns the last element, or `null` if the sequence is empty.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.lastOrNull(): T? {\n val iterator = iterator()\n if (!iterator.hasNext())\n return null\n var last = iterator.next()\n while (iterator.hasNext())\n last = iterator.next()\n return last\n}\n\n/**\n * Returns the last element matching the given [predicate], or `null` if no such element was found.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.lastOrNull(predicate: (T) -> Boolean): T? {\n var last: T? = null\n for (element in this) {\n if (predicate(element))\n last = element\n }\n return last\n}\n\n/**\n * Returns the single element, or throws an exception if the sequence is empty or has more than one element.\n *\n * The operation is _terminal_.\n */\npublic fun <T> Sequence<T>.single(): T {\n val iterator = iterator()\n if (!iterator.hasNext())\n throw NoSuchElementException("Sequence is empty.")\n val single = iterator.next()\n if (iterator.hasNext())\n throw IllegalArgumentException("Sequence has more than one element.")\n return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.single(predicate: (T) -> Boolean): T {\n var single: T? = null\n var found = false\n for (element in this) {\n if (predicate(element))\n if (found) throw IllegalArgumentException("Sequence contains more than one matching element.")\n single = element\n found = true\n }\n if (!found) throw NoSuchElementException("Sequence contains no element matching the predicate.")\n @Suppress("UNCHECKED_CAST")\n return single as T\n}\n\n/**\n * Returns single element, or `null` if the sequence is empty or has more than one element.\n *\n * The

```

operation is _terminal_.\n *\npublic fun <T> Sequence<T>.singleOrNull(): T? {\n    val iterator = iterator()\n    if (!iterator.hasNext())\n        return null\n    val single = iterator.next()\n    if (iterator.hasNext())\n        return null\n    return single\n}\n\n/**\n * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.\n *\n * The operation is _terminal_.\n *\npublic inline fun <T> Sequence<T>.singleOrNull(predicate: (T) -> Boolean): T? {\n    var single: T? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single = element\n            found = true\n        }\n    }\n    if (!found) return null\n    return single\n}\n\n/**\n * Returns a sequence containing all elements except first [n] elements.\n *\n * The operation is _intermediate_ and _stateless_.\n *\n * @throws IllegalArgumentException if [n] is negative.\n *\n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic fun <T> Sequence<T>.drop(n: Int): Sequence<T> {\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    return when {\n        n == 0 -> this\n        this is DropTakeSequence -> this.drop(n)\n        else -> DropSequence(this, n)\n    }\n}\n\n/**\n * Returns a sequence containing all elements except first elements that satisfy the given [predicate].\n *\n * The operation is _intermediate_ and _stateless_.\n *\n * @sample
samples.collections.Collections.Transformations.drop\n *\npublic fun <T> Sequence<T>.dropWhile(predicate: (T) -> Boolean): Sequence<T> {\n    return DropWhileSequence(this, predicate)\n}\n\n/**\n * Returns a sequence containing only elements matching the given [predicate].\n *\n * The operation is _intermediate_ and _stateless_.\n *\npublic fun <T> Sequence<T>.filter(predicate: (T) -> Boolean): Sequence<T> {\n    return FilteringSequence(this, true, predicate)\n}\n\n/**\n * Returns a sequence containing only elements matching the given [predicate].\n *\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n *\n * The operation is _intermediate_ and _stateless_.\n *\npublic fun <T> Sequence<T>.filterIndexed(predicate: (index: Int, T) -> Boolean): Sequence<T> {\n    // TODO: Rewrite with generalized MapFilterIndexingSequence\n    return TransformingSequence(FilteringSequence(IndexingSequence(this), true, { predicate(it.index, it.value) }), { it.value })\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n *\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n *\n * The operation is _terminal_.\n *\npublic inline fun <T, C : MutableCollection<in T>> Sequence<T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {\n    for (index, element) in this\n        if (predicate(index, element)) destination.add(element)\n    return destination\n}\n\n/**\n * Returns a sequence containing all elements that are instances of specified type parameter R.\n *\n * The operation is _intermediate_ and _stateless_.\n *\npublic inline fun <reified R> Sequence<*>.filterIsInstance(): Sequence<@kotlin.internal.NoInfer R> {\n    @Suppress("UNCHECKED_CAST")\n    return filter { it is R } as Sequence<R>\n}\n\n/**\n * Appends all elements that are instances of specified type parameter R to the given [destination].\n *\n * The operation is _terminal_.\n *\npublic inline fun <reified R, C : MutableCollection<in R>> Sequence<*>.filterIsInstanceTo(destination: C): C {\n    for (element in this) if (element is R)\n        destination.add(element)\n    return destination\n}\n\n/**\n * Returns a sequence containing all elements not matching the given [predicate].\n *\n * The operation is _intermediate_ and _stateless_.\n *\npublic fun <T> Sequence<T>.filterNot(predicate: (T) -> Boolean): Sequence<T> {\n    return FilteringSequence(this, false, predicate)\n}\n\n/**\n * Returns a sequence containing all elements that are not `null`.\n *\n * The operation is _intermediate_ and _stateless_.\n *\npublic fun <T : Any> Sequence<T?>.filterNotNull(): Sequence<T> {\n    @Suppress("UNCHECKED_CAST")\n    return filterNot { it == null } as Sequence<T>\n}\n\n/**\n * Appends all elements that are not `null` to the given [destination].\n *\n * The operation is _terminal_.\n *\npublic fun <C : MutableCollection<in T>, T : Any> Sequence<T?>.filterNotNullTo(destination: C): C {\n    for (element in this) if (element != null)\n        destination.add(element)\n    return destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n *\n * The operation is _terminal_.\n *\npublic inline fun <T, C : MutableCollection<in T>> Sequence<T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {\n    for (element in this) if (!predicate(element))\n        destination.add(element)\n    return destination\n}\n\n/**\n * Appends all

```

elements matching the given [predicate] to the given [destination].
`inline fun <T, C : MutableCollection<in T>> Sequence<T>.filterTo(destination: C, predicate: (T) -> Boolean): C`
 {
 for (element in this) if (predicate(element)) destination.add(element)
 return destination
 }
 Returns a sequence containing first [n] elements.
 The operation is `_intermediate_` and `_stateless_`.
 @throws `IllegalArgumentException` if [n] is negative.
 @sample
`samples.collections.Collections.Transformations.take`
`inline fun <T> Sequence<T>.take(n: Int): Sequence<T>`
 {
 require(n >= 0) { "Requested element count \$n is less than zero." }
 return when {
 n == 0 -> emptySequence()
 this is DropTakeSequence -> this.take(n)
 else -> TakeSequence(this, n)
 }
 }
 Returns a sequence containing first elements satisfying the given [predicate].
 The operation is `_intermediate_` and `_stateless_`.
 @sample `samples.collections.Collections.Transformations.take`
`inline fun <T> Sequence<T>.takeWhile(predicate: (T) -> Boolean): Sequence<T>`
 {
 return TakeWhileSequence(this, predicate)
 }
 Returns a sequence that yields elements of this sequence sorted according to their natural sort order.
 The sort is `_stable_`. It means that equal elements preserve their order relative to each other after sorting.
 The operation is `_intermediate_` and `_stateful_`.
`inline fun <T : Comparable<T>> Sequence<T>.sorted(): Sequence<T>`
 {
 return object : Sequence<T> {
 override fun iterator(): Iterator<T> {
 val sortedList = this@sorted.toMutableList()
 sortedList.sort()
 return sortedList.iterator()
 }
 }
 }
 Returns a sequence that yields elements of this sequence sorted according to natural sort order of the value returned by specified [selector] function.
 The sort is `_stable_`. It means that equal elements preserve their order relative to each other after sorting.
 The operation is `_intermediate_` and `_stateful_`.
`inline fun <T, R : Comparable<R>> Sequence<T>.sortedBy(crossinline selector: (T) -> R?): Sequence<T>`
 {
 return sortedWith(compareBy(selector))
 }
 Returns a sequence that yields elements of this sequence sorted descending according to natural sort order of the value returned by specified [selector] function.
 The sort is `_stable_`. It means that equal elements preserve their order relative to each other after sorting.
 The operation is `_intermediate_` and `_stateful_`.
`inline fun <T, R : Comparable<R>> Sequence<T>.sortedByDescending(crossinline selector: (T) -> R?): Sequence<T>`
 {
 return sortedWith(compareByDescending(selector))
 }
 Returns a sequence that yields elements of this sequence sorted descending according to their natural sort order.
 The sort is `_stable_`. It means that equal elements preserve their order relative to each other after sorting.
 The operation is `_intermediate_` and `_stateful_`.
`inline fun <T : Comparable<T>> Sequence<T>.sortedDescending(): Sequence<T>`
 {
 return sortedWith(reverseOrder())
 }
 Returns a sequence that yields elements of this sequence sorted according to the specified [comparator].
 The sort is `_stable_`. It means that equal elements preserve their order relative to each other after sorting.
 The operation is `_intermediate_` and `_stateful_`.
`inline fun <T> Sequence<T>.sortedWith(comparator: Comparator<in T>): Sequence<T>`
 {
 return object : Sequence<T> {
 override fun iterator(): Iterator<T> {
 val sortedList = this@sortedWith.toMutableList()
 sortedList.sortWith(comparator)
 return sortedList.iterator()
 }
 }
 }
 Returns a [Map] containing key-value pairs provided by [transform] function applied to elements of the given sequence.
 If any of two pairs would have the same key the last one gets added to the map.
 The returned map preserves the entry iteration order of the original sequence.
 The operation is `_terminal_`.
`inline fun <T, K, V> Sequence<T>.associate(transform: (T) -> Pair<K, V>): Map<K, V>`
 {
 return associateTo(LinkedHashMap<K, V>(), transform)
 }
 Returns a [Map] containing the elements from the given sequence indexed by the key returned from [keySelector] function applied to each element.
 If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 The returned map preserves the entry iteration order of the original sequence.
 The operation is `_terminal_`.
`inline fun <T, K> Sequence<T>.associateBy(keySelector: (T) -> K): Map<K, T>`
 {
 return associateByTo(LinkedHashMap<K, T>(), keySelector)
 }
 Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given sequence.
 If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* \n * The returned map preserves the entry iteration order of the original sequence.\n * \n * The operation is _terminal_.\n * \n public inline fun <T, K, V> Sequence<T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {\n return associateByTo(LinkedHashMap<K, V>(), keySelector, valueTransform)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given sequence\n * and value is the element itself.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The operation is _terminal_.\n * \n public inline fun <T, K, M : MutableMap<in K, in T>> Sequence<T>.associateByTo(destination: M, keySelector: (T) -> K): M {\n for (element in this) {\n destination.put(keySelector(element), element)\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given sequence.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The operation is _terminal_.\n * \n public inline fun <T, K, V, M : MutableMap<in K, in V>> Sequence<T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {\n for (element in this) {\n destination.put(keySelector(element), valueTransform(element))\n }\n return destination\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given sequence.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n * \n * The operation is _terminal_.\n * \n public inline fun <T, K, V, M : MutableMap<in K, in V>> Sequence<T>.associateTo(destination: M, transform: (T) -> Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n return destination\n}\n\n/**\n * Returns a [Map] where keys are elements from the given sequence and values are\n * produced by the [valueSelector] function applied to each element.\n * \n * If any two elements are equal, the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original sequence.\n * \n * The operation is _terminal_.\n * \n * @sample samples.collections.Collections.Transformations.associateWith\n * \n @SinceKotlin("1.3")\n public inline fun <K, V> Sequence<K>.associateWith(valueSelector: (K) -> V): Map<K, V> {\n val result = LinkedHashMap<K, V>()\n return associateWithTo(result, valueSelector)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value pairs for each element of the given sequence,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * \n * If any two elements are equal, the last one overwrites the former value in the map.\n * \n * The operation is _terminal_.\n * \n @SinceKotlin("1.3")\n public inline fun <K, V, M : MutableMap<in K, in V>> Sequence<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {\n for (element in this) {\n destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Appends all elements to the given [destination] collection.\n * \n * The operation is _terminal_.\n * \n public fun <T, C : MutableCollection<in T>> Sequence<T>.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return destination\n}\n\n/**\n * Returns a [HashSet] of all elements.\n * \n * The operation is _terminal_.\n * \n public fun <T> Sequence<T>.toHashSet(): HashSet<T> {\n return toCollection(HashSet<T>())\n}\n\n/**\n * Returns a [List] containing all elements.\n * \n * The operation is _terminal_.\n * \n public fun <T> Sequence<T>.toList(): List<T> {\n return this.toMutableList().optimizeReadOnlyList()\n}\n\n/**\n * Returns a [MutableList] filled with all elements of this sequence.\n * \n * The operation is _terminal_.\n * \n public fun <T> Sequence<T>.toMutableList(): MutableList<T> {\n return toCollection(ArrayList<T>())\n}\n\n/**\n * Returns a [Set] of all elements.\n * \n * The returned set preserves the element iteration order of the original sequence.\n * \n * The operation is _terminal_.\n * \n public fun <T> Sequence<T>.toSet(): Set<T> {\n return toCollection(LinkedHashSet<T>()).optimizeReadOnlySet()\n}\n\n/**\n * Returns a single sequence of all elements from results of [transform] function being invoked on each element of original sequence.\n * \n * The operation is _intermediate_ and _stateless_.\n * \n public fun <T, R> Sequence<T>.flatMap(transform: (T) -> Sequence<R>): Sequence<R> {\n return FlatteningSequence(this, transform, { it.iterator() })\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each element of original sequence, to the given


```

[destination].\n * \n * The operation is _terminal_.\n * \n *\npublic inline fun <T, R, C : MutableCollection<in R>>
Sequence<T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {\n  for (element in this) {\n    val
list = transform(element)\n    destination.addAll(list)\n  }\n  return destination\n}\n\n/**\n * Groups elements of
the original sequence by the key returned by the given [keySelector] function\n * applied to each element and
returns a map where each group key is associated with a list of corresponding elements.\n * \n * The returned map
preserves the entry iteration order of the keys produced from the original sequence.\n * \n * The operation is
_terminal_.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n * \n *\npublic inline fun <T,
K> Sequence<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {\n  return groupByTo(LinkedHashMap<K,
MutableList<T>>(), keySelector)\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to
each element of the original sequence\n * by the key returned by the given [keySelector] function applied to the
element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The
returned map preserves the entry iteration order of the keys produced from the original sequence.\n * \n * The
operation is _terminal_.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n * \n *\npublic inline fun <T, K, V>
Sequence<T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups elements
of the original sequence by the key returned by the given [keySelector] function\n * applied to each element and
puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The
[destination] map.\n * \n * The operation is _terminal_.\n * \n * @sample
samples.collections.Collections.Transformations.groupBy\n * \n *\npublic inline fun <T, K, M : MutableMap<in K,
MutableList<T>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K): M {\n  for (element in this)
{\n    val key = keySelector(element)\n    val list = destination.getOrPut(key) { ArrayList<T>() }\n
list.add(element)\n  }\n  return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function
applied to each element of the original sequence\n * by the key returned by the given [keySelector] function applied
to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n
* \n * @return The [destination] map.\n * \n * The operation is _terminal_.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n * \n *\npublic inline fun <T, K, V, M :
MutableMap<in K, MutableList<V>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K,
valueTransform: (T) -> V): M {\n  for (element in this) {\n    val key = keySelector(element)\n    val list =
destination.getOrPut(key) { ArrayList<V>() }\n    list.add(valueTransform(element))\n  }\n  return
destination\n}\n\n/**\n * Creates a [Grouping] source from a sequence to be used later with one of group-and-fold
operations\n * using the specified [keySelector] function to extract a key from each element.\n * \n * The operation is
_intermediate_ and _stateless_.\n * \n * @sample samples.collections.Grouping.groupingByEachCount\n
* \n *\n@SinceKotlin("1.1")\npublic inline fun <T, K> Sequence<T>.groupingBy(crossinline keySelector: (T) -> K):
Grouping<T, K> {\n  return object : Grouping<T, K> {\n    override fun sourceIterator(): Iterator<T> =
this@groupingBy.iterator()\n    override fun keyOf(element: T): K = keySelector(element)\n  }\n}\n\n/**\n * Returns a sequence containing the results of applying the given [transform] function\n * to each element in the
original sequence.\n * \n * The operation is _intermediate_ and _stateless_.\n * \n * @sample
samples.collections.Collections.Transformations.map\n * \n *\npublic fun <T, R> Sequence<T>.map(transform: (T) ->
R): Sequence<R> {\n  return TransformingSequence(this, transform)\n}\n\n/**\n * Returns a sequence containing
the results of applying the given [transform] function\n * to each element and its index in the original sequence.\n
* \n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of
the transform applied to the element.\n * \n * The operation is _intermediate_ and _stateless_.\n * \n *\npublic fun <T,
R> Sequence<T>.mapIndexed(transform: (index: Int, T) -> R): Sequence<R> {\n  return
TransformingIndexedSequence(this, transform)\n}\n\n/**\n * Returns a sequence containing only the non-null
results of applying the given [transform] function\n * to each element and its index in the original sequence.\n
* \n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of
the transform applied to the element.\n * \n * The operation is _intermediate_ and _stateless_.\n * \n *\npublic fun <T, R

```

```

: Any> Sequence<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): Sequence<R> {
    return TransformingIndexedSequence(this, transform).filterNotNull()
}
    * Applies the given [transform] function to each element and its index in the original sequence
    * and appends only the non-null results to the given [destination].
    * @param [transform] function that takes the index of an element and the element itself
    * and returns the result of the transform applied to the element.
    * The operation is _terminal_.
    * public inline fun <T, R : Any, C : MutableCollection<in R>> Sequence<T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {
    forEachIndexed { index, element -> transform(index, element)?.let { destination.add(it) } }
    return destination
}
    * Applies the given [transform] function to each element and its index in the original sequence
    * and appends the results to the given [destination].
    * @param [transform] function that takes the index of an element and the element itself
    * and returns the result of the transform applied to the element.
    * The operation is _terminal_.
    * public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(checkIndexOverflow(index++), item))
    return destination
}
    * Returns a sequence containing only the non-null results of applying the given [transform] function
    * to each element in the original sequence.
    * The operation is _intermediate_ and _stateless_.
    * public fun <T, R : Any> Sequence<T>.mapNotNull(transform: (T) -> R?): Sequence<R> {
    return TransformingSequence(this, transform).filterNotNull()
}
    * Applies the given [transform] function to each element in the original sequence
    * and appends only the non-null results to the given [destination].
    * The operation is _terminal_.
    * public inline fun <T, R : Any, C : MutableCollection<in R>> Sequence<T>.mapNotNullTo(destination: C, transform: (T) -> R?): C {
    forEach { element -> transform(element)?.let { destination.add(it) } }
    return destination
}
    * Applies the given [transform] function to each element of the original sequence
    * and appends the results to the given [destination].
    * The operation is _terminal_.
    * public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.mapTo(destination: C, transform: (T) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
    * Returns a sequence that wraps each element of the original sequence
    * into an [IndexedValue] containing the index of that element and the element itself.
    * The operation is _intermediate_ and _stateless_.
    * public fun <T> Sequence<T>.withIndex(): Sequence<IndexedValue<T>> {
    return IndexingSequence(this)
}
    * Returns a sequence containing only distinct elements from the given sequence.
    * The elements in the resulting sequence are in the same order as they were in the source sequence.
    * The operation is _intermediate_ and _stateful_.
    * public fun <T> Sequence<T>.distinct(): Sequence<T> {
    return this.distinctBy { it }
}
    * Returns a sequence containing only elements from the given sequence
    * having distinct keys returned by the given [selector] function.
    * The elements in the resulting sequence are in the same order as they were in the source sequence.
    * The operation is _intermediate_ and _stateful_.
    * public fun <T, K> Sequence<T>.distinctBy(selector: (T) -> K): Sequence<T> {
    return DistinctSequence(this, selector)
}
    * Returns a mutable set containing all distinct elements from the given sequence.
    * The returned set preserves the element iteration order of the original sequence.
    * The operation is _terminal_.
    * public fun <T> Sequence<T>.toMutableSet(): MutableSet<T> {
    val set = LinkedHashSet<T>()
    for (item in this)
        set.add(item)
    return set
}
    * Returns `true` if all elements match the given [predicate].
    * The operation is _terminal_.
    * @sample samples.collections.Collections.Aggregates.all
    * public inline fun <T> Sequence<T>.all(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
    * Returns `true` if sequence has at least one element.
    * The operation is _terminal_.
    * @sample samples.collections.Collections.Aggregates.any
    * public inline fun <T> Sequence<T>.any(): Boolean {
    return iterator().hasNext()
}
    * Returns `true` if at least one element matches the given [predicate].
    * The operation is _terminal_.
    * @sample samples.collections.Collections.Aggregates.anyWithPredicate
    * public inline fun <T> Sequence<T>.any(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}
    * Returns the number of elements in this sequence.
    * The operation is _terminal_.
    * public fun <T> Sequence<T>.count(): Int {
    var count = 0
    for (element in this)

```

```

checkCountOverflow(++count)\n    return count\n}\n\n/**\n * Returns the number of elements matching the given
[predicate].\n *\n * The operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.count(predicate: (T) ->
Boolean): Int {\n    var count = 0\n    for (element in this) if (predicate(element)) checkCountOverflow(++count)\n    return count\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to right
to current accumulator value and each element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T, R>
Sequence<T>.fold(initial: R, operation: (acc: R, T) -> R): R {\n    var accumulator = initial\n    for (element in this)\n        accumulator = operation(accumulator, element)\n    return accumulator\n}\n\n/**\n * Accumulates value starting
with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element
with its index in the original sequence.\n * @param [operation] function that takes the index of an element, current
accumulator value\n * and the element itself, and calculates the next accumulator value.\n *\n * The operation is
_terminal_.\n */\npublic inline fun <T, R> Sequence<T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) ->
R): R {\n    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator =
operation(checkIndexOverflow(index++), accumulator, element)\n    return accumulator\n}\n\n/**\n * Performs the
given [action] on each element.\n *\n * The operation is _terminal_.\n */\npublic inline fun <T>
Sequence<T>.forEach(action: (T) -> Unit): Unit {\n    for (element in this) action(element)\n}\n\n/**\n * Performs
the given [action] on each element, providing sequential index with the element.\n * @param [action] function that
takes the index of an element and the element itself\n * and performs the desired action on the element.\n *\n * The
operation is _terminal_.\n */\npublic inline fun <T> Sequence<T>.forEachIndexed(action: (index: Int, T) -> Unit):
Unit {\n    var index = 0\n    for (item in this) action(checkIndexOverflow(index++), item)\n}\n\n/**\n * Returns the
largest element or `null` if there are no elements.\n *\n * If any of elements is `NaN` returns `NaN`.\n *\n * The
operation is _terminal_.\n */\n@SinceKotlin("1.1")\npublic fun Sequence<Double>.max(): Double? {\n    val
iterator = iterator()\n    if (!iterator.hasNext()) return null\n    var max = iterator.next()\n    if (max.isNaN()) return
max\n    while (iterator.hasNext()) {\n        val e = iterator.next()\n        if (e.isNaN()) return e\n        if (max < e) max
= e\n    }\n    return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n *\n * If any
of elements is `NaN` returns `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin("1.1")\npublic fun
Sequence<Float>.max(): Float? {\n    val iterator = iterator()\n    if (!iterator.hasNext()) return null\n    var max =
iterator.next()\n    if (max.isNaN()) return max\n    while (iterator.hasNext()) {\n        val e = iterator.next()\n        if
(e.isNaN()) return e\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the largest element or
`null` if there are no elements.\n *\n * The operation is _terminal_.\n */\npublic fun <T : Comparable<T>>
Sequence<T>.max(): T? {\n    val iterator = iterator()\n    if (!iterator.hasNext()) return null\n    var max =
iterator.next()\n    while (iterator.hasNext()) {\n        val e = iterator.next()\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there
are no elements.\n *\n * The operation is _terminal_.\n */\n\n * @sample
samples.collections.Collections.Aggregates.maxBy\n */\npublic inline fun <T, R : Comparable<R>>
Sequence<T>.maxBy(selector: (T) -> R): T? {\n    val iterator = iterator()\n    if (!iterator.hasNext()) return null\n    var
maxElem = iterator.next()\n    if (!iterator.hasNext()) return maxElem\n    var maxValue = selector(maxElem)\n    do {\n        val e = iterator.next()\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    } while (iterator.hasNext())\n    return maxElem\n}\n\n/**\n * Returns the first element
having the largest value according to the provided [comparator] or `null` if there are no elements.\n *\n * The
operation is _terminal_.\n */\npublic fun <T> Sequence<T>.maxWith(comparator: Comparator<in T>): T? {\n    val
iterator = iterator()\n    if (!iterator.hasNext()) return null\n    var max = iterator.next()\n    while (iterator.hasNext())\n        {\n            val e = iterator.next()\n            if (comparator.compare(max, e) < 0) max = e\n        }\n    return max\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n *\n * If any of elements is `NaN` returns `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin("1.1")\npublic fun Sequence<Double>.min(): Double? {\n    val
iterator = iterator()\n    if (!iterator.hasNext()) return null\n    var min = iterator.next()\n    if (min.isNaN()) return
min\n    while (iterator.hasNext()) {\n        val e = iterator.next()\n        if (e.isNaN()) return e\n        if (min > e) min
= e\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n *\n * If any
of elements is `NaN` returns `NaN`.\n *\n * The operation is _terminal_.\n */\n@SinceKotlin("1.1")\npublic fun

```

```

Sequence<Float>.min(): Float? {\n  val iterator = iterator()\n  if (!iterator.hasNext()) return null\n  var min =
iterator.next()\n  if (min.isNaN()) return min\n  while (iterator.hasNext()) {\n    val e = iterator.next()\n    if
(e.isNaN()) return e\n    if (min > e) min = e\n  }\n  return min\n}\n\n/**\n * Returns the smallest element or
`null` if there are no elements.\n *\n * The operation is _terminal_.\n */\npublic fun <T : Comparable<T>>
Sequence<T>.min(): T? {\n  val iterator = iterator()\n  if (!iterator.hasNext()) return null\n  var min =
iterator.next()\n  while (iterator.hasNext()) {\n    val e = iterator.next()\n    if (min > e) min = e\n  }\n  return
min\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no
elements.\n *\n * The operation is _terminal_.\n */\n * @sample
samples.collections.Collections.Aggregates.minBy\n */\npublic inline fun <T, R : Comparable<R>>
Sequence<T>.minBy(selector: (T) -> R): T? {\n  val iterator = iterator()\n  if (!iterator.hasNext()) return null\n
var minElem = iterator.next()\n  if (!iterator.hasNext()) return minElem\n  var minValue = selector(minElem)\n
do {\n    val e = iterator.next()\n    val v = selector(e)\n    if (minValue > v) {\n      minElem = e\n
minValue = v\n    }\n  } while (iterator.hasNext())\n  return minElem\n}\n\n/**\n * Returns the first element
having the smallest value according to the provided [comparator] or `null` if there are no elements.\n *\n * The
operation is _terminal_.\n */\npublic fun <T> Sequence<T>.minWith(comparator: Comparator<in T>): T? {\n  val
iterator = iterator()\n  if (!iterator.hasNext()) return null\n  var min = iterator.next()\n  while (iterator.hasNext())
{\n    val e = iterator.next()\n    if (comparator.compare(min, e) > 0) min = e\n  }\n  return min\n}\n\n/**\n *
Returns `true` if the sequence has no elements.\n *\n * The operation is _terminal_.\n */\n * @sample
samples.collections.Collections.Aggregates.none\n */\npublic fun <T> Sequence<T>.none(): Boolean {\n  return
!iterator().hasNext()\n}\n\n/**\n * Returns `true` if no elements match the given [predicate].\n *\n * The operation is
_terminal_.\n */\n * @sample samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun
<T> Sequence<T>.none(predicate: (T) -> Boolean): Boolean {\n  for (element in this) if (predicate(element))
return false\n  return true\n}\n\n/**\n * Returns a sequence which performs the given [action] on each element of
the original sequence as they pass through it.\n *\n * The operation is _intermediate_ and _stateless_.\n
*/\n * @SinceKotlin("1.1")\n */\npublic fun <T> Sequence<T>.onEach(action: (T) -> Unit): Sequence<T> {\n  return
map {\n    action(it)\n    it\n  }\n}\n\n/**\n * Accumulates value starting with the first element and applying
[operation] from left to right to current accumulator value and each element.\n *\n * The operation is _terminal_.\n
*/\n * @public inline fun <S, T : S> Sequence<T>.reduce(operation: (acc: S, T) -> S): S {\n  val iterator =
this.iterator()\n  if (!iterator.hasNext()) throw UnsupportedOperationException("Empty sequence can't be
reduced.")\n  var accumulator: S = iterator.next()\n  while (iterator.hasNext()) {\n    accumulator =
operation(accumulator, iterator.next())\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with
the first element and applying [operation] from left to right\n * to current accumulator value and each element with
its index in the original sequence.\n *\n * @param [operation] function that takes the index of an element, current
accumulator value\n * and the element itself and calculates the next accumulator value.\n *\n * The operation is
_terminal_.\n */\n * @public inline fun <S, T : S> Sequence<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S
{\n  val iterator = this.iterator()\n  if (!iterator.hasNext()) throw UnsupportedOperationException("Empty
sequence can't be reduced.")\n  var index = 1\n  var accumulator: S = iterator.next()\n  while (iterator.hasNext())
{\n    accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())\n  }\n  return
accumulator\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in
the sequence.\n *\n * The operation is _terminal_.\n */\n * @public inline fun <T> Sequence<T>.sumBy(selector: (T) ->
Int): Int {\n  var sum: Int = 0\n  for (element in this) {\n    sum += selector(element)\n  }\n  return
sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function applied to each element in the
sequence.\n *\n * The operation is _terminal_.\n */\n * @public inline fun <T> Sequence<T>.sumByDouble(selector:
(T) -> Double): Double {\n  var sum: Double = 0.0\n  for (element in this) {\n    sum += selector(element)\n
}\n  return sum\n}\n\n/**\n * Returns an original collection containing all the non-`null` elements, throwing an
[IllegalArgumentException] if there are any `null` elements.\n *\n * The operation is _intermediate_ and
_stateless_.\n */\n * @public fun <T : Any> Sequence<T?>.requireNonNulls(): Sequence<T> {\n  return map { it?:
throw IllegalArgumentException("null element found in $this.") }\n}\n\n/**\n * Splits this sequence into a

```

sequence of lists each not exceeding the given [size].
The last list in the resulting sequence may have less elements than the given [size].
@param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this sequence.
The operation is `_intermediate_` and `_stateful_`.
@sample samples.collections.Collections.Transformations.chunked

```
*\n@SinceKotlin("1.2")\npublic fun <T> Sequence<T>.chunked(size: Int): Sequence<List<T>> {\n    return windowed(size, size, partialWindows = true)\n}\n\n/**\n * Splits this sequence into several lists each not exceeding the given [size] and applies the given [transform] function to an each.\n * @return sequence of results of the [transform] applied to an each list.\n * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * The last list may have less elements than the given [size].\n * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this sequence.\n * The operation is _intermediate_ and _stateful_.  
@sample samples.text.Strings.chunkedTransform\n\n*\n@SinceKotlin("1.2")\npublic fun <T, R> Sequence<T>.chunked(size: Int, transform: (List<T>) -> R): Sequence<R> {\n    return windowed(size, size, partialWindows = true, transform = transform)\n}\n\n/**\n * Returns a sequence containing all elements of the original sequence without the first occurrence of the given [element].\n * The operation is _intermediate_ and _stateless_.  
\npublic operator fun <T> Sequence<T>.minus(element: T): Sequence<T> {\n    return object: Sequence<T> {\n        override fun iterator(): Iterator<T> {\n            var removed = false\n            return this@minus.filter { if (!removed && it == element) { removed = true; false } else true }.iterator()\n        }\n    }\n}\n\n/**\n * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] array.\n * Note that the source sequence and the array being subtracted are iterated only when an `iterator` is requested from the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n * The [elements] array may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.  
The operation is _intermediate_ and _stateful_.  
\npublic operator fun <T> Sequence<T>.minus(elements: Array<out T>): Sequence<T> {\n    if (elements.isEmpty()) return this\n    return object: Sequence<T> {\n        override fun iterator(): Iterator<T> {\n            val other = elements.toHashSet()\n            return this@minus.filterNot { it in other }.iterator()\n        }\n    }\n}\n\n/**\n * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] collection.\n * Note that the source sequence and the collection being subtracted are iterated only when an `iterator` is requested from the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.  
The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.  
The operation is _intermediate_ and _stateful_.  
\npublic operator fun <T> Sequence<T>.minus(elements: Iterable<T>): Sequence<T> {\n    return object: Sequence<T> {\n        override fun iterator(): Iterator<T> {\n            val other = elements.convertToSetForSetOperation()\n            if (other.isEmpty()) return this@minus.iterator()\n            else return this@minus.filterNot { it in other }.iterator()\n        }\n    }\n}\n\n/**\n * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] sequence.  
Note that the source sequence and the sequence being subtracted are iterated only when an `iterator` is requested from the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.  
The operation is _intermediate_ for this sequence and _terminal_ and _stateful_ for the [elements] sequence.  
The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.  
\npublic operator fun <T> Sequence<T>.minus(elements: Sequence<T>): Sequence<T> {\n    return object: Sequence<T> {\n        override fun iterator(): Iterator<T> {\n            val other = elements.toHashSet()\n            if (other.isEmpty()) return this@minus.iterator()\n            else return this@minus.filterNot { it in other }.iterator()\n        }\n    }\n}\n\n/**\n * Returns a sequence containing all elements of the original sequence without the first occurrence of the given [element].\n * The operation is
```

```

_intermediate_ and _stateless_.\n *\n@kotlin.internal.InlineOnly\npublic inline fun <T>
Sequence<T>.minusElement(element: T): Sequence<T> {\n    return minus(element)\n}\n\n/**\n * Splits the
original sequence into pair of lists,\n * where *first* list contains elements for which [predicate] yielded `true`,\n *
while *second* list contains elements for which [predicate] yielded `false`.\n *\n * The operation is _terminal_.\n
*\npublic inline fun <T> Sequence<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {\n    val first
= ArrayList<T>()\n    val second = ArrayList<T>()\n    for (element in this) {\n        if (predicate(element)) {\n
first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return Pair(first,
second)\n}\n\n/**\n * Returns a sequence containing all elements of the original sequence and then the given
[element].\n *\n * The operation is _intermediate_ and _stateless_.\n *\npublic operator fun <T>
Sequence<T>.plus(element: T): Sequence<T> {\n    return sequenceOf(this,
sequenceOf(element)).flatten()\n}\n\n/**\n * Returns a sequence containing all elements of original sequence and
then all elements of the given [elements] array.\n *\n * Note that the source sequence and the array being added are
iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between
successive calls to `iterator` may affect the result.\n *\n * The operation is _intermediate_ and _stateless_.\n
*\npublic operator fun <T> Sequence<T>.plus(elements: Array<out T>): Sequence<T> {\n    return
this.plus(elements.asList())\n}\n\n/**\n * Returns a sequence containing all elements of original sequence and
then all elements of the given [elements] collection.\n *\n * Note that the source sequence and the collection being added
are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between
successive calls to `iterator` may affect the result.\n *\n * The operation is _intermediate_ and _stateless_.\n
*\npublic operator fun <T> Sequence<T>.plus(elements: Iterable<T>): Sequence<T> {\n    return
sequenceOf(this,
elements.asSequence()).flatten()\n}\n\n/**\n * Returns a sequence containing all elements of original sequence and
then all elements of the given [elements] sequence.\n *\n * Note that the source sequence and the sequence being
added are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them
between successive calls to `iterator` may affect the result.\n *\n * The operation is _intermediate_ and _stateless_.\n
*\npublic operator fun <T> Sequence<T>.plus(elements: Sequence<T>): Sequence<T> {\n    return
sequenceOf(this, elements).flatten()\n}\n\n/**\n * Returns a sequence containing all elements of the original
sequence and then the given [element].\n *\n * The operation is _intermediate_ and _stateless_.\n
*\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.plusElement(element: T): Sequence<T> {\n
return plus(element)\n}\n\n/**\n * Returns a sequence of snapshots of the window of the given [size]\n * sliding
along this sequence with the given [step], where each\n * snapshot is a list.\n *\n * Several last lists may have less
elements than the given [size].\n *\n * Both [size] and [step] must be positive and can be greater than the number of
elements in this sequence.\n *\n * @param size the number of elements to take in each window\n * @param step the
number of elements to move the window forward by on an each step, by default 1\n * @param partialWindows
controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial
windows won't be preserved\n *\n * @sample samples.collections.Sequences.Transformations.takeWindows\n
*\n@SinceKotlin("1.2")\npublic fun <T> Sequence<T>.windowed(size: Int, step: Int = 1, partialWindows:
Boolean = false): Sequence<List<T>> {\n    return windowedSequence(size, step, partialWindows, reuseBuffer =
false)\n}\n\n/**\n * Returns a sequence of results of applying the given [transform] function to\n * an each list
representing a view over the window of the given [size]\n * sliding along this sequence with the given [step].\n *\n *
Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.\n * You
should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * Several last lists may
have less elements than the given [size].\n *\n * Both [size] and [step] must be positive and can be greater than the
number of elements in this sequence.\n *\n * @param size the number of elements to take in each window\n * @param
step the number of elements to move the window forward by on an each step, by default 1\n * @param
partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which
means partial windows won't be preserved\n *\n * @sample
samples.collections.Sequences.Transformations.averageWindows\n
*\n@SinceKotlin("1.2")\npublic fun <T, R>
Sequence<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (List<T>) -> R):

```

```

Sequence<R> {\n    return windowedSequence(size, step, partialWindows, reuseBuffer =
true).map(transform)\n}\n\n/**\n * Returns a sequence of values built from the elements of `this` sequence and the
[other] sequence with the same index.\n * The resulting sequence ends as soon as the shortest input sequence ends.\n
*\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample
samples.collections.Sequences.Transformations.zip\n *\npublic infix fun <T, R> Sequence<T>.zip(other:
Sequence<R>): Sequence<Pair<T, R>> {\n    return MergingSequence(this, other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n *
Returns a sequence of values built from the elements of `this` sequence and the [other] sequence with the same
index\n * using the provided [transform] function applied to each pair of elements.\n * The resulting sequence ends
as soon as the shortest input sequence ends.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n *
@sample samples.collections.Sequences.Transformations.zipWithTransform\n *\npublic fun <T, R, V>
Sequence<T>.zip(other: Sequence<R>, transform: (a: T, b: R) -> V): Sequence<V> {\n    return
MergingSequence(this, other, transform)\n}\n\n/**\n * Returns a sequence of pairs of each two adjacent elements in
this sequence.\n * \n * The returned sequence is empty if this sequence contains less than two elements.\n *\n * The
operation is _intermediate_ and _stateless_.\n * \n * @sample
samples.collections.Collections.Transformations.zipWithNext\n *\n@SinceKotlin("1.2")\npublic fun <T>
Sequence<T>.zipWithNext(): Sequence<Pair<T, T>> {\n    return zipWithNext { a, b -> a to b }\n}\n\n/**\n *
Returns a sequence containing the results of applying the given [transform] function\n * to an each pair of two
adjacent elements in this sequence.\n *\n * The returned sequence is empty if this sequence contains less than two
elements.\n *\n * The operation is _intermediate_ and _stateless_.\n * \n * @sample
samples.collections.Collections.Transformations.zipWithNextToFindDeltas\n *\n@SinceKotlin("1.2")\npublic
fun <T, R> Sequence<T>.zipWithNext(transform: (a: T, b: T) -> R): Sequence<R> {\n    return sequence result@
{\n        val iterator = iterator()\n        if (!iterator.hasNext()) return @result\n        var current = iterator.next()\n
while (iterator.hasNext()) {\n            val next = iterator.next()\n            yield(transform(current, next))\n
current = next\n        }\n    }\n}\n\n/**\n * Appends the string from all the elements separated using [separator] and
using the given [prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-
negative value of [limit], in which case only the first [limit]\n * elements will be appended, followed by the
[truncated] string (which defaults to "...").\n *\n * The operation is _terminal_.\n * \n * @sample
samples.collections.Collections.Transformations.joinTo\n *\npublic fun <T, A : Appendable>
Sequence<T>.joinTo(buffer: A, separator: CharSequence = "\", \"", prefix: CharSequence = "\"", postfix:
CharSequence = "\"", limit: Int = -1, truncated: CharSequence = "...\", transform: ((T) -> CharSequence)? = null): A
{\n    buffer.append(prefix)\n    var count = 0\n    for (element in this) {\n        if (++count > 1)
buffer.append(separator)\n        if (limit < 0 || count <= limit) {\n            buffer.appendElement(element, transform)\n
        } else break\n    }\n    if (limit >= 0 && count > limit) buffer.append(truncated)\n    buffer.append(postfix)\n
return buffer\n}\n\n/**\n * Creates a string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied.\n * \n * If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit]\n * elements will be appended, followed by the [truncated] string (which
defaults to "...").\n *\n * The operation is _terminal_.\n * \n * @sample
samples.collections.Collections.Transformations.joinToString\n *\npublic fun <T>
Sequence<T>.joinToString(separator: CharSequence = "\", \"", prefix: CharSequence = "\"", postfix: CharSequence =
\"", limit: Int = -1, truncated: CharSequence = "...\", transform: ((T) -> CharSequence)? = null): String {\n    return
joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()\n}\n\n/**\n * Creates an
[Iterable] instance that wraps the original sequence returning its elements when being iterated.\n *\npublic fun <T>
Sequence<T>.asIterable(): Iterable<T> {\n    return Iterable { this.iterator() }\n}\n\n/**\n * Returns this sequence as
a [Sequence].\n *\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.asSequence(): Sequence<T>
{\n    return this\n}\n\n/**\n * Returns an average value of elements in the sequence.\n *\n * The operation is
_terminal_.\n *\n@kotlin.jvm.JvmName("averageOfByte")\npublic fun Sequence<Byte>.average(): Double {\n
var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) {\n        sum += element\n
checkCountOverflow(++count)\n    }\n    return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns

```

```

an average value of elements in the sequence.\n *\n * The operation is _terminal_.\n
*\n@kotlin.jvm.JvmName("\averageOfShort")\npublic fun Sequence<Short>.average(): Double {\n    var sum:
Double = 0.0\n    var count: Int = 0\n    for (element in this) {\n        sum += element\n
checkCountOverflow(++count)\n    }\n    return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns
an average value of elements in the sequence.\n *\n * The operation is _terminal_.\n
*\n@kotlin.jvm.JvmName("\averageOfInt")\npublic fun Sequence<Int>.average(): Double {\n    var sum: Double
= 0.0\n    var count: Int = 0\n    for (element in this) {\n        sum += element\n        checkCountOverflow(++count)\n
    }\n    return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns an average value of elements in the
sequence.\n *\n * The operation is _terminal_.\n *\n@kotlin.jvm.JvmName("\averageOfLong")\npublic fun
Sequence<Long>.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) {\n
sum += element\n        checkCountOverflow(++count)\n    }\n    return if (count == 0) Double.NaN else sum /
count\n}\n\n/**\n * Returns an average value of elements in the sequence.\n *\n * The operation is _terminal_.\n
*\n@kotlin.jvm.JvmName("\averageOfFloat")\npublic fun Sequence<Float>.average(): Double {\n    var sum:
Double = 0.0\n    var count: Int = 0\n    for (element in this) {\n        sum += element\n
checkCountOverflow(++count)\n    }\n    return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns
an average value of elements in the sequence.\n *\n * The operation is _terminal_.\n
*\n@kotlin.jvm.JvmName("\averageOfDouble")\npublic fun Sequence<Double>.average(): Double {\n    var sum:
Double = 0.0\n    var count: Int = 0\n    for (element in this) {\n        sum += element\n
checkCountOverflow(++count)\n    }\n    return if (count == 0) Double.NaN else sum / count\n}\n\n/**\n * Returns
the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n
*\n@kotlin.jvm.JvmName("\sumOfByte")\npublic fun Sequence<Byte>.sum(): Int {\n    var sum: Int = 0\n    for
(element in this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all elements in the
sequence.\n *\n * The operation is _terminal_.\n *\n@kotlin.jvm.JvmName("\sumOfShort")\npublic fun
Sequence<Short>.sum(): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum += element\n    }\n    return
sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n
*\n@kotlin.jvm.JvmName("\sumOfInt")\npublic fun Sequence<Int>.sum(): Int {\n    var sum: Int = 0\n    for
(element in this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all elements in the
sequence.\n *\n * The operation is _terminal_.\n *\n@kotlin.jvm.JvmName("\sumOfLong")\npublic fun
Sequence<Long>.sum(): Long {\n    var sum: Long = 0L\n    for (element in this) {\n        sum += element\n    }\n
return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n *\n * The operation is _terminal_.\n
*\n@kotlin.jvm.JvmName("\sumOfFloat")\npublic fun Sequence<Float>.sum(): Float {\n    var sum: Float = 0.0f\n
for (element in this) {\n        sum += element\n    }\n    return sum\n}\n\n/**\n * Returns the sum of all elements in
the sequence.\n *\n * The operation is _terminal_.\n *\n@kotlin.jvm.JvmName("\sumOfDouble")\npublic fun
Sequence<Double>.sum(): Double {\n    var sum: Double = 0.0\n    for (element in this) {\n        sum += element\n
    }\n    return sum\n}\n\n",null,"*\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\n
*\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("\SetsKt")\n\npackage
kotlin.collections\n\n/\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n\nimport kotlin.random.*\nimport
kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns a set containing all elements of the original
set except the given [element].\n *\n * The returned set preserves the element iteration order of the original set.\n
*\n * \npublic operator fun <T> Set<T>.minus(element: T): Set<T> {\n    val result =
LinkedHashSet<T>(mapCapacity(size))\n    var removed = false\n    return this.filterTo(result) { if (!removed && it
== element) { removed = true; false } else true }\n}\n\n/**\n * Returns a set containing all elements of the original
set except the elements contained in the given [elements] array.\n *\n * The returned set preserves the element
iteration order of the original set.\n *\n * The [elements] array may be converted to a [HashSet] to speed up the
operation, thus the elements are required to have\n * a correct and stable implementation of `hashCode()` that

```


doesn't change between successive invocations.

```

public operator fun <T> Set<T>.minus(elements: Array<out T>): Set<T> {
    val result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

```

* Returns a set containing all elements of the original set except the elements contained in the given [elements] collection.

* The returned set preserves the element iteration order of the original set.

* The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.

```

public operator fun <T> Set<T>.minus(elements: Iterable<T>): Set<T> {
    val other = elements.convertToSetForSetOperationWith(this)
    if (other.isEmpty()) return this.toSet()
    if (other is Set) return this.filterNotTo(LinkedHashSet<T>()) { it in other }
    val result = LinkedHashSet<T>(this)
    result.removeAll(other)
    return result
}

```

* Returns a set containing all elements of the original set except the elements contained in the given [elements] sequence.

* The returned set preserves the element iteration order of the original set.

* The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.

```

public operator fun <T> Set<T>.minus(elements: Sequence<T>): Set<T> {
    val result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

```

* Returns a set containing all elements of the original set except the given [element].

* The returned set preserves the element iteration order of the original set.

```

@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.minusElement(element: T): Set<T> {
    return minus(element)
}

```

* Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.

* The returned set preserves the element iteration order of the original set.

```

public operator fun <T> Set<T>.plus(element: T): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(size + 1))
    result.addAll(this)
    result.add(element)
    return result
}

```

* Returns a set containing all elements of the original set and the given [elements] array, which aren't already in this set.

* The returned set preserves the element iteration order of the original set.

```

public operator fun <T> Set<T>.plus(elements: Array<out T>): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(this.size + elements.size))
    result.addAll(this)
    result.addAll(elements)
    return result
}

```

* Returns a set containing all elements of the original set and the given [elements] collection, which aren't already in this set.

* The returned set preserves the element iteration order of the original set.

```

public operator fun <T> Set<T>.plus(elements: Iterable<T>): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(elements.collectionSizeOrNull()?.let { this.size + it } ?: this.size * 2))
    result.addAll(this)
    result.addAll(elements)
    return result
}

```

* Returns a set containing all elements of the original set and the given [elements] sequence, which aren't already in this set.

* The returned set preserves the element iteration order of the original set.

```

public operator fun <T> Set<T>.plus(elements: Sequence<T>): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(this.size * 2))
    result.addAll(this)
    result.addAll(elements)
    return result
}

```

* Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.

* The returned set preserves the element iteration order of the original set.

```

@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.plusElement(element: T): Set<T> {
    return plus(element)
}

```

"/>**
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

```

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")
package kotlin.text
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
import kotlin.random.*

```

* Returns a character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char sequence.

* @sample samples.collections.Collections.Elements.elementAt

```

public expect fun CharSequence.elementAt(index: Int): Char

```

* Returns a character at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this char sequence.

* @sample samples.collections.Collections.Elements.elementAtOrElse

```

@kotlin.internal.InlineOnly
public inline fun CharSequence.elementAtOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index

```

```

<= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns a character at the given [index] or `null` if
the [index] is out of bounds of this char sequence.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n * \n @kotlin.internal.InlineOnly\npublic inline fun
CharSequence.elementAtOrNull(index: Int): Char? {\n    return this.getOrNull(index)\n}\n\n/**\n * Returns the first
character matching the given [predicate], or `null` if no such character was found.\n
*\n @kotlin.internal.InlineOnly\npublic inline fun CharSequence.find(predicate: (Char) -> Boolean): Char? {\n
return firstOrNull(predicate)\n}\n\n/**\n * Returns the last character matching the given [predicate], or `null` if no
such character was found.\n * \n @kotlin.internal.InlineOnly\npublic inline fun CharSequence.findLast(predicate:
(Char) -> Boolean): Char? {\n    return lastOrNull(predicate)\n}\n\n/**\n * Returns first character.\n * @throws
[NoSuchElementException] if the char sequence is empty.\n * \n public fun CharSequence.first(): Char {\n    if
(isEmpty())\n        throw NoSuchElementException("Char sequence is empty.")\n    return this[0]\n}\n\n/**\n *
Returns the first character matching the given [predicate].\n * @throws [NoSuchElementException] if no such
character is found.\n * \n public inline fun CharSequence.first(predicate: (Char) -> Boolean): Char {\n    for (element
in this) if (predicate(element)) return element\n    throw NoSuchElementException("Char sequence contains no
character matching the predicate.")\n}\n\n/**\n * Returns the first character, or `null` if the char sequence is
empty.\n * \n public fun CharSequence.firstOrNull(): Char? {\n    return if (isEmpty()) null else this[0]\n}\n\n/**\n *
Returns the first character matching the given [predicate], or `null` if character was not found.\n * \n public inline fun
CharSequence.firstOrNull(predicate: (Char) -> Boolean): Char? {\n    for (element in this) if (predicate(element))
return element\n    return null\n}\n\n/**\n * Returns a character at the given [index] or the result of calling the
[defaultValue] function if the [index] is out of bounds of this char sequence.\n
*\n @kotlin.internal.InlineOnly\npublic inline fun CharSequence.getOrNull(index: Int, defaultValue: (Int) -> Char):
Char {\n    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns a
character at the given [index] or `null` if the [index] is out of bounds of this char sequence.\n * \n public fun
CharSequence.getOrNull(index: Int): Char? {\n    return if (index >= 0 && index <= lastIndex) get(index) else
null\n}\n\n/**\n * Returns index of the first character matching the given [predicate], or -1 if the char sequence does
not contain such character.\n * \n public inline fun CharSequence.indexOfFirst(predicate: (Char) -> Boolean): Int {\n
for (index in indices) {\n    if (predicate(this[index])) {\n        return index\n    }\n }\n return -
1\n}\n\n/**\n * Returns index of the last character matching the given [predicate], or -1 if the char sequence does
not contain such character.\n * \n public inline fun CharSequence.indexOfLast(predicate: (Char) -> Boolean): Int {\n
for (index in indices.reversed()) {\n    if (predicate(this[index])) {\n        return index\n    }\n }\n return -
1\n}\n\n/**\n * Returns the last character.\n * @throws [NoSuchElementException] if the char sequence is empty.\n
*\n public fun CharSequence.last(): Char {\n    if (isEmpty())\n        throw NoSuchElementException("Char
sequence is empty.")\n    return this[lastIndex]\n}\n\n/**\n * Returns the last character matching the given
[predicate].\n * @throws [NoSuchElementException] if no such character is found.\n * \n public inline fun
CharSequence.last(predicate: (Char) -> Boolean): Char {\n    for (index in this.indices.reversed()) {\n        val
element = this[index]\n        if (predicate(element)) return element\n    }\n    throw NoSuchElementException("Char
sequence contains no character matching the predicate.")\n}\n\n/**\n * Returns the last character, or `null` if the
char sequence is empty.\n * \n public fun CharSequence.lastOrNull(): Char? {\n    return if (isEmpty()) null else
this[length - 1]\n}\n\n/**\n * Returns the last character matching the given [predicate], or `null` if no such character
was found.\n * \n public inline fun CharSequence.lastOrNull(predicate: (Char) -> Boolean): Char? {\n    for (index in
this.indices.reversed()) {\n        val element = this[index]\n        if (predicate(element)) return element\n    }\n
return null\n}\n\n/**\n * Returns a random character from this char sequence.\n * \n * @throws
NoSuchElementException if this char sequence is empty.\n
*\n @kotlin.SinceKotlin("1.3")\n @kotlin.internal.InlineOnly\npublic inline fun CharSequence.random(): Char {\n
return random(Random)\n}\n\n/**\n * Returns a random character from this char sequence using the specified
source of randomness.\n * \n * @throws NoSuchElementException if this char sequence is empty.\n
*\n @kotlin.SinceKotlin("1.3")\n public fun CharSequence.random(random: Random): Char {\n    if (isEmpty())\n        throw NoSuchElementException("Char sequence is empty.")\n    return get(random.nextInt(length))\n}\n\n/**\n *

```

```

Returns the single character, or throws an exception if the char sequence is empty or has more than one character.
*^public fun CharSequence.single(): Char {
    return when (length) {
        0 -> throw
        NoSuchElementException("Char sequence is empty.")
        1 -> this[0]
        else -> throw
        IllegalArgumentException("Char sequence has more than one element.")
    }
}
* Returns the single character matching the given [predicate], or throws exception if there is no or more than one matching character.
*^public inline fun CharSequence.single(predicate: (Char) -> Boolean): Char {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw
            IllegalArgumentException("Char sequence contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Char sequence contains no character matching the predicate.")
    @SuppressWarnings("UNCHECKED_CAST")
    return single as Char
}
* Returns single character, or `null` if the char sequence is empty or has more than one character.
*^public fun CharSequence.singleOrNull(): Char? {
    return if (length == 1) this[0] else null
}
* Returns the single character matching the given [predicate], or `null` if character was not found or more than one character was found.
*^public inline fun CharSequence.singleOrNull(predicate: (Char) -> Boolean): Char? {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}
* Returns a subsequence of this char sequence with the first [n] characters removed.
* @throws IllegalArgumentException if [n] is negative.
* @sample samples.text.Strings.drop
*^public fun CharSequence.drop(n: Int): CharSequence {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return subSequence(n.coerceAtMost(length), length)
}
* Returns a string with the first [n] characters removed.
* @throws IllegalArgumentException if [n] is negative.
* @sample samples.text.Strings.drop
*^public fun String.drop(n: Int): String {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return substring(n.coerceAtMost(length))
}
* Returns a subsequence of this char sequence with the last [n] characters removed.
* @throws IllegalArgumentException if [n] is negative.
* @sample samples.text.Strings.drop
*^public fun CharSequence.dropLast(n: Int): CharSequence {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return take((length - n).coerceAtLeast(0))
}
* Returns a string with the last [n] characters removed.
* @throws IllegalArgumentException if [n] is negative.
* @sample samples.text.Strings.drop
*^public fun String.dropLast(n: Int): String {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return take((length - n).coerceAtLeast(0))
}
* Returns a subsequence of this char sequence containing all characters except last characters that satisfy the given [predicate].
* @sample samples.text.Strings.drop
*^public inline fun CharSequence.dropLastWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return subSequence(0, index + 1)
        }
    }
    return ""
}
* Returns a string containing all characters except last characters that satisfy the given [predicate].
* @sample samples.text.Strings.drop
*^public inline fun String.dropLastWhile(predicate: (Char) -> Boolean): String {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return substring(0, index + 1)
        }
    }
    return ""
}
* Returns a subsequence of this char sequence containing all characters except first characters that satisfy the given [predicate].
* @sample samples.text.Strings.drop
*^public inline fun CharSequence.dropWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices) {
        if (!predicate(this[index])) {
            return subSequence(index, length)
        }
    }
    return ""
}
* Returns a string containing all characters except first characters that satisfy the given [predicate].
* @sample samples.text.Strings.drop
*^public inline fun String.dropWhile(predicate: (Char) -> Boolean): String {
    for (index in this.indices) {
        if (!predicate(this[index])) {
            return substring(index)
        }
    }
    return ""
}
* Returns a char sequence containing only those characters from the original char sequence that match the given [predicate].
*^public inline fun CharSequence.filter(predicate: (Char) -> Boolean): CharSequence {
    return filterTo(StringBuilder(), predicate)
}
* Returns a string containing only those characters from the original string that match the given [predicate].
*^public inline fun String.filter(predicate: (Char) -> Boolean): String {
    return filterTo(StringBuilder(), predicate).toString()
}
* Returns a char

```

```

sequence containing only those characters from the original char sequence that match the given [predicate].\n *
@param [predicate] function that takes the index of a character and the character itself\n * and returns the result of
predicate evaluation on the character.\n */\npublic inline fun CharSequence.filterIndexed(predicate: (index: Int,
Char) -> Boolean): CharSequence {\n    return filterIndexedTo(StringBuilder(), predicate)\n}\n\n/**\n * Returns a
string containing only those characters from the original string that match the given [predicate].\n * @param
[predicate] function that takes the index of a character and the character itself\n * and returns the result of predicate
evaluation on the character.\n */\npublic inline fun String.filterIndexed(predicate: (index: Int, Char) -> Boolean):
String {\n    return filterIndexedTo(StringBuilder(), predicate).toString()\n}\n\n/**\n * Appends all characters
matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of a
character and the character itself\n * and returns the result of predicate evaluation on the character.\n */\npublic
inline fun <C : Appendable> CharSequence.filterIndexedTo(destination: C, predicate: (index: Int, Char) ->
Boolean): C {\n    forEachIndexed { index, element ->\n        if (predicate(index, element))
destination.append(element)\n    }\n    return destination\n}\n\n/**\n * Returns a char sequence containing only
those characters from the original char sequence that do not match the given [predicate].\n */\npublic inline fun
CharSequence.filterNot(predicate: (Char) -> Boolean): CharSequence {\n    return filterNotTo(StringBuilder(),
predicate)\n}\n\n/**\n * Returns a string containing only those characters from the original string that do not match
the given [predicate].\n */\npublic inline fun String.filterNot(predicate: (Char) -> Boolean): String {\n    return
filterNotTo(StringBuilder(), predicate).toString()\n}\n\n/**\n * Appends all characters not matching the given
[predicate] to the given [destination].\n */\npublic inline fun <C : Appendable>
CharSequence.filterNotTo(destination: C, predicate: (Char) -> Boolean): C {\n    for (element in this) if
(!predicate(element)) destination.append(element)\n    return destination\n}\n\n/**\n * Appends all characters
matching the given [predicate] to the given [destination].\n */\npublic inline fun <C : Appendable>
CharSequence.filterTo(destination: C, predicate: (Char) -> Boolean): C {\n    for (index in 0 until length) {\n        val
element = get(index)\n        if (predicate(element)) destination.append(element)\n    }\n    return
destination\n}\n\n/**\n * Returns a char sequence containing characters of the original char sequence at the
specified range of [indices].\n */\npublic fun CharSequence.slice(indices: IntRange): CharSequence {\n    if
(indices.isEmpty()) return ""\n    return subSequence(indices)\n}\n\n/**\n * Returns a string containing characters
of the original string at the specified range of [indices].\n */\npublic fun String.slice(indices: IntRange): String {\n
if (indices.isEmpty()) return ""\n    return substring(indices)\n}\n\n/**\n * Returns a char sequence containing
characters of the original char sequence at specified [indices].\n */\npublic fun CharSequence.slice(indices:
Iterable<Int>): CharSequence {\n    val size = indices.collectionSizeOrDefault(10)\n    if (size == 0) return ""\n    val
result = StringBuilder(size)\n    for (i in indices) {\n        result.append(get(i))\n    }\n    return result\n}\n\n/**\n * Returns a string containing characters of the original string at specified [indices].\n
\n */\n\n/**\n * @kotlin.internal.InlineOnly\n */\npublic inline fun String.slice(indices: Iterable<Int>): String {\n    return (this as
CharSequence).slice(indices).toString()\n}\n\n/**\n * Returns a subsequence of this char sequence containing the
first [n] characters from this char sequence, or the entire char sequence if this char sequence is shorter.\n * \n *
\n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.take\n */\npublic fun
CharSequence.take(n: Int): CharSequence {\n    require(n >= 0) { "Requested character count $n is less than zero." }\n
return subSequence(0, n.coerceAtMost(length))\n}\n\n/**\n * Returns a string containing the first [n]
characters from this string, or the entire string if this string is shorter.\n * \n * \n * @throws IllegalArgumentException if
[n] is negative.\n * \n * @sample samples.text.Strings.take\n */\npublic fun String.take(n: Int): String {\n    require(n
>= 0) { "Requested character count $n is less than zero." }\n    return substring(0,
n.coerceAtMost(length))\n}\n\n/**\n * Returns a subsequence of this char sequence containing the last [n]
characters from this char sequence, or the entire char sequence if this char sequence is shorter.\n * \n * \n * @throws
IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.take\n */\npublic fun
CharSequence.takeLast(n: Int): CharSequence {\n    require(n >= 0) { "Requested character count $n is less than
zero." }\n    val length = length\n    return subSequence(length - n.coerceAtMost(length), length)\n}\n\n/**\n *
Returns a string containing the last [n] characters from this string, or the entire string if this string is shorter.\n * \n *
\n */

```

```

@throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.text.Strings.take\n * \n\npublic fun
String.takeLast(n: Int): String {\n    require(n >= 0) {"Requested character count $n is less than zero."}\n    val
length = length\n    return substring(length - n.coerceAtMost(length))\n}\n\n/**\n * Returns a subsequence of this
char sequence containing last characters that satisfy the given [predicate].\n * \n * @sample
samples.text.Strings.take\n * \n\npublic inline fun CharSequence.takeLastWhile(predicate: (Char) -> Boolean):
CharSequence {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) {\n            return
subSequence(index + 1, length)\n        }\n    }\n    return subSequence(0, length)\n}\n\n/**\n * Returns a string
containing last characters that satisfy the given [predicate].\n * \n * @sample samples.text.Strings.take\n * \n\npublic
inline fun String.takeLastWhile(predicate: (Char) -> Boolean): String {\n    for (index in lastIndex downTo 0) {\n
        if (!predicate(this[index])) {\n            return substring(index + 1)\n        }\n    }\n    return this\n}\n\n/**\n * Returns
a subsequence of this char sequence containing the first characters that satisfy the given [predicate].\n * \n *
@sample samples.text.Strings.take\n * \n\npublic inline fun CharSequence.takeWhile(predicate: (Char) -> Boolean):
CharSequence {\n    for (index in 0 until length)\n        if (!predicate(get(index))) {\n            return
subSequence(0,
index)\n        }\n    return subSequence(0, length)\n}\n\n/**\n * Returns a string containing the first characters that
satisfy the given [predicate].\n * \n * @sample samples.text.Strings.take\n * \n\npublic inline fun
String.takeWhile(predicate: (Char) -> Boolean): String {\n    for (index in 0 until length)\n        if
(!predicate(get(index))) {\n            return substring(0, index)\n        }\n    return this\n}\n\n/**\n * Returns a char
sequence with characters in reversed order.\n * \n\npublic fun CharSequence.reversed(): CharSequence {\n    return
StringBuilder(this).reverse()\n}\n\n/**\n * Returns a string with characters in reversed order.\n * \n\n@kotlin.internal.InlineOnly\npublic inline fun String.reversed(): String {\n    return (this as
CharSequence).reversed().toString()\n}\n\n/**\n * Returns a [Map] containing key-value pairs provided by
[transform] function\n * applied to characters of the given char sequence.\n * \n * If any of two pairs would have the
same key the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the
original char sequence.\n * \n\npublic inline fun <K, V> CharSequence.associate(transform: (Char) -> Pair<K, V>):
Map<K, V> {\n    val capacity = mapCapacity(length).coerceAtLeast(16)\n    return
associateTo(LinkedHashMap<K, V>(capacity), transform)\n}\n\n/**\n * Returns a [Map] containing the characters
from the given char sequence indexed by the key\n * returned from [keySelector] function applied to each
character.\n * \n * If any two characters would have the same key returned by [keySelector] the last one gets added
to the map.\n * \n * The returned map preserves the entry iteration order of the original char sequence.\n * \n\npublic
inline fun <K> CharSequence.associateBy(keySelector: (Char) -> K): Map<K, Char> {\n    val capacity =
mapCapacity(length).coerceAtLeast(16)\n    return associateByTo(LinkedHashMap<K, Char>(capacity),
keySelector)\n}\n\n/**\n * Returns a [Map] containing the values provided by [valueTransform] and indexed by
[keySelector] functions applied to characters of the given char sequence.\n * \n * If any two characters would have
the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the
entry iteration order of the original char sequence.\n * \n\npublic inline fun <K, V>
CharSequence.associateBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {\n    val capacity
= mapCapacity(length).coerceAtLeast(16)\n    return associateByTo(LinkedHashMap<K, V>(capacity),
keySelector, valueTransform)\n}\n\n/**\n * Populates and returns the [destination] mutable map with key-value
pairs,\n * where key is provided by the [keySelector] function applied to each character of the given char sequence\n *
and value is the character itself.\n * \n * If any two characters would have the same key returned by [keySelector]
the last one gets added to the map.\n * \n\npublic inline fun <K, M : MutableMap<in K, in Char>>
CharSequence.associateByTo(destination: M, keySelector: (Char) -> K): M {\n    for (element in this) {\n
        destination.put(keySelector(element), element)\n    }\n    return destination\n}\n\n/**\n * Populates and returns the
[destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and
value is provided by the [valueTransform] function applied to characters of the given char sequence.\n * \n * If any
two characters would have the same key returned by [keySelector] the last one gets added to the map.\n * \n\npublic
inline fun <K, V, M : MutableMap<in K, in V>> CharSequence.associateByTo(destination: M, keySelector: (Char)
-> K, valueTransform: (Char) -> V): M {\n    for (element in this) {\n        destination.put(keySelector(element),

```

```

valueTransform(element))\n } return destination\n}\n\n/**\n * Populates and returns the [destination] mutable
map with key-value pairs\n * provided by [transform] function applied to each character of the given char
sequence.\n * \n * If any of two pairs would have the same key the last one gets added to the map.\n */\npublic
inline fun <K, V, M : MutableMap<in K, in V>> CharSequence.associateTo(destination: M, transform: (Char) ->
Pair<K, V>): M {\n for (element in this) {\n destination += transform(element)\n }\n return
destination\n}\n\n/**\n * Returns a [Map] where keys are characters from the given char sequence and values are\n
* produced by the [valueSelector] function applied to each character.\n * \n * If any two characters are equal, the
last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original char
sequence.\n * \n * @sample samples.text.Strings.associateWith\n */\n@SinceKotlin("1.3")\npublic inline fun <V>
CharSequence.associateWith(valueSelector: (Char) -> V): Map<Char, V> {\n val result = LinkedHashMap<Char,
V>(mapCapacity(length).coerceAtLeast(16))\n return associateWithTo(result, valueSelector)\n}\n\n/**\n *
Populates and returns the [destination] mutable map with key-value pairs for each character of the given char
sequence,\n * where key is the character itself and value is provided by the [valueSelector] function applied to that
key.\n * \n * If any two characters are equal, the last one overwrites the former value in the map.\n
*/\n@SinceKotlin("1.3")\npublic inline fun <V, M : MutableMap<in Char, in V>>
CharSequence.associateWithTo(destination: M, valueSelector: (Char) -> V): M {\n for (element in this) {\n
destination.put(element, valueSelector(element))\n }\n return destination\n}\n\n/**\n * Appends all characters to
the given [destination] collection.\n */\npublic fun <C : MutableCollection<in Char>>
CharSequence.toCollection(destination: C): C {\n for (item in this) {\n destination.add(item)\n }\n return
destination\n}\n\n/**\n * Returns a [HashSet] of all characters.\n */\npublic fun CharSequence.toHashSet():
HashSet<Char> {\n return toCollection(HashSet<Char>(mapCapacity(length)))\n}\n\n/**\n * Returns a [List]
containing all characters.\n */\npublic fun CharSequence.toList(): List<Char> {\n return when (length) {\n 0 -
> emptyList()\n 1 -> listOf(this[0])\n else -> this.toMutableList()\n }\n}\n\n/**\n * Returns a
[MutableList] filled with all characters of this char sequence.\n */\npublic fun CharSequence.toMutableList():
MutableList<Char> {\n return toCollection(ArrayList<Char>(length))\n}\n\n/**\n * Returns a [Set] of all
characters.\n * \n * The returned set preserves the element iteration order of the original char sequence.\n
*/\npublic fun CharSequence.toSet(): Set<Char> {\n return when (length) {\n 0 -> emptySet()\n 1 -> setOf(this[0])\n
else -> toCollection(LinkedHashSet<Char>(mapCapacity(length)))\n }\n}\n\n/**\n * Returns a single list of all
elements yielded from results of [transform] function being invoked on each character of original char sequence.\n
*/\npublic inline fun <R> CharSequence.flatMap(transform: (Char) -> Iterable<R>): List<R> {\n return
flatMapTo(ArrayList<R>(), transform)\n}\n\n/**\n * Appends all elements yielded from results of [transform]
function being invoked on each character of original char sequence, to the given [destination].\n */\npublic inline
fun <R, C : MutableCollection<in R>> CharSequence.flatMapTo(destination: C, transform: (Char) -> Iterable<R>):
C {\n for (element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return
destination\n}\n\n/**\n * Groups characters of the original char sequence by the key returned by the given
[keySelector] function\n * applied to each character and returns a map where each group key is associated with a list
of corresponding characters.\n * \n * The returned map preserves the entry iteration order of the keys produced from
the original char sequence.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */\npublic
inline fun <K> CharSequence.groupBy(keySelector: (Char) -> K): Map<K, List<Char>> {\n return
groupByTo(LinkedHashMap<K, MutableList<Char>>(), keySelector)\n}\n\n/**\n * Groups values returned by the
[valueTransform] function applied to each character of the original char sequence\n * by the key returned by the
given [keySelector] function applied to the character\n * and returns a map where each group key is associated with
a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced
from the original char sequence.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */\npublic inline fun <K, V>
CharSequence.groupBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>> {\n return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups
characters of the original char sequence by the key returned by the given [keySelector] function\n * applied to each

```

```

character and puts to the [destination] map each group key associated with a list of corresponding characters.\n * \n
* @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*/\npublic inline fun <K, M : MutableMap<in K, MutableList<Char>>> CharSequence.groupByTo(destination: M,\n
keySelector: (Char) -> K): M {\n    for (element in this) {\n        val key = keySelector(element)\n        val list =\n
destination.getOrPut(key) { ArrayList<Char>() }\n        list.add(element)\n    }\n    return destination\n}\n\n/**\n * Groups values returned by the [valueTransform] function applied to each character of the original char sequence\n *
by the key returned by the given [keySelector] function applied to the character\n * and puts to the [destination] map\n
each group key associated with a list of corresponding values.\n * \n * @return The [destination] map.\n * \n *
@sample samples.collections.Collections.Transformations.groupByKeyAndValues\n */\npublic inline fun <K, V,\n
M : MutableMap<in K, MutableList<V>>> CharSequence.groupByTo(destination: M, keySelector: (Char) -> K,\n
valueTransform: (Char) -> V): M {\n    for (element in this) {\n        val key = keySelector(element)\n        val list =\n
destination.getOrPut(key) { ArrayList<V>() }\n        list.add(valueTransform(element))\n    }\n    return\n
destination\n}\n\n/**\n * Creates a [Grouping] source from a char sequence to be used later with one of group-and-\n
fold operations\n * using the specified [keySelector] function to extract a key from each character.\n * \n * @sample\n
samples.collections.Grouping.groupingByEachCount\n */\n@SinceKotlin("1.1")\npublic inline fun <K>\n
CharSequence.groupingBy(crossinline keySelector: (Char) -> K): Grouping<Char, K> {\n    return object :\n
Grouping<Char, K> {\n        override fun sourceIterator(): Iterator<Char> = this@groupingBy.iterator()\n
        override fun keyOf(element: Char): K = keySelector(element)\n    }\n}\n\n/**\n * Returns a list containing the\n
results of applying the given [transform] function\n * to each character in the original char sequence.\n * \n *
@sample samples.text.Strings.map\n */\npublic inline fun <R> CharSequence.map(transform: (Char) -> R):\n
List<R> {\n    return mapTo(ArrayList<R>(length), transform)\n}\n\n/**\n * Returns a list containing the results of\n
applying the given [transform] function\n * to each character and its index in the original char sequence.\n *
@param [transform] function that takes the index of a character and the character itself\n * and returns the result of\n
the transform applied to the character.\n */\npublic inline fun <R> CharSequence.mapIndexed(transform: (index:\n
Int, Char) -> R): List<R> {\n    return mapIndexedTo(ArrayList<R>(length), transform)\n}\n\n/**\n * Returns a list\n
containing only the non-null results of applying the given [transform] function\n * to each character and its index in\n
the original char sequence.\n * @param [transform] function that takes the index of a character and the character\n
itself\n * and returns the result of the transform applied to the character.\n */\npublic inline fun <R : Any>\n
CharSequence.mapIndexedNotNull(transform: (index: Int, Char) -> R?): List<R> {\n    return\n
mapIndexedNotNullTo(ArrayList<R>(), transform)\n}\n\n/**\n * Applies the given [transform] function to each\n
character and its index in the original char sequence\n * and appends only the non-null results to the given\n
[destination].\n * @param [transform] function that takes the index of a character and the character itself\n * and\n
returns the result of the transform applied to the character.\n */\npublic inline fun <R : Any, C :  

MutableCollection<in R>> CharSequence.mapIndexedNotNullTo(destination: C, transform: (index: Int, Char) ->\n
R?): C {\n    forEachIndexed { index, element -> transform(index, element)?.let { destination.add(it) } }\n    return\n
destination\n}\n\n/**\n * Applies the given [transform] function\n * to each character and its index in the original char\n
sequence\n * and appends the results to the given [destination].\n * @param [transform] function that takes the\n
index of a character and the character itself\n * and returns the result of the transform applied to the character.\n
*/\npublic inline fun <R, C : MutableCollection<in R>> CharSequence.mapIndexedTo(destination: C, transform:\n
(index: Int, Char) -> R): C {\n    var index = 0\n    for (item in this)\n        destination.add(transform(index++,\n
item))\n    return destination\n}\n\n/**\n * Returns a list containing only the non-null results of applying the given\n
[transform] function\n * to each character in the original char sequence.\n */\npublic inline fun <R : Any>\n
CharSequence.mapNotNull(transform: (Char) -> R?): List<R> {\n    return mapNotNullTo(ArrayList<R>(),\n
transform)\n}\n\n/**\n * Applies the given [transform] function to each character in the original char sequence\n *
and appends only the non-null results to the given [destination].\n */\npublic inline fun <R : Any, C :  

MutableCollection<in R>> CharSequence.mapNotNullTo(destination: C, transform: (Char) -> R?): C {\n    forEach\n
{ element -> transform(element)?.let { destination.add(it) } }\n    return destination\n}\n\n/**\n * Applies the given\n
[transform] function to each character of the original char sequence\n * and appends the results to the given

```

```

[destination].\n *\npublic inline fun <R, C : MutableCollection<in R>> CharSequence.mapTo(destination: C,
transform: (Char) -> R): C {\n for (item in this)\n destination.add(transform(item))\n return
destination}\n\n/**\n * Returns a lazy [Iterable] that wraps each character of the original char sequence\n * into an
[IndexValue] containing the index of that character and the character itself.\n *\npublic fun
CharSequence.withIndex(): Iterable<IndexedValue<Char>> {\n return IndexingIterable { iterator() }\n}\n\n/**\n *
Returns `true` if all characters match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.all\n *\npublic inline fun CharSequence.all(predicate: (Char) ->
Boolean): Boolean {\n for (element in this) if (!predicate(element)) return false\n return true}\n\n/**\n *
Returns `true` if char sequence has at least one character.\n * \n * @sample
samples.collections.Collections.Aggregates.any\n *\npublic fun CharSequence.any(): Boolean {\n return
!isEmpty()\n}\n\n/**\n * Returns `true` if at least one character matches the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.anyWithPredicate\n *\npublic inline fun CharSequence.any(predicate:
(Char) -> Boolean): Boolean {\n for (element in this) if (predicate(element)) return true\n return
false}\n\n/**\n * Returns the length of this char sequence.\n *\n@kotlin.internal.InlineOnly\npublic inline fun
CharSequence.count(): Int {\n return length}\n\n/**\n * Returns the number of characters matching the given
[predicate].\n *\npublic inline fun CharSequence.count(predicate: (Char) -> Boolean): Int {\n var count = 0\n
for (element in this) if (predicate(element)) ++count\n return count}\n\n/**\n * Accumulates value starting with
[initial] value and applying [operation] from left to right to current accumulator value and each character.\n
*\npublic inline fun <R> CharSequence.fold(initial: R, operation: (acc: R, Char) -> R): R {\n var accumulator =
initial\n for (element in this) accumulator = operation(accumulator, element)\n return accumulator}\n\n/**\n *
Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator
value and each character with its index in the original char sequence.\n * @param [operation] function that takes the
index of a character, current accumulator value\n * and the character itself, and calculates the next accumulator
value.\n *\npublic inline fun <R> CharSequence.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R):
R {\n var index = 0\n var accumulator = initial\n for (element in this) accumulator = operation(index++,
accumulator, element)\n return accumulator}\n\n/**\n * Accumulates value starting with [initial] value and
applying [operation] from right to left to each character and current accumulator value.\n *\npublic inline fun <R>
CharSequence.foldRight(initial: R, operation: (Char, acc: R) -> R): R {\n var index = lastIndex\n var
accumulator = initial\n while (index >= 0) {\n accumulator = operation(get(index--), accumulator)\n }\n
return accumulator}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right
to left\n * to each character with its index in the original char sequence and current accumulator value.\n * @param
[operation] function that takes the index of a character, the character itself\n * and current accumulator value, and
calculates the next accumulator value.\n *\npublic inline fun <R> CharSequence.foldRightIndexed(initial: R,
operation: (index: Int, Char, acc: R) -> R): R {\n var index = lastIndex\n var accumulator = initial\n while
(index >= 0) {\n accumulator = operation(index, get(index), accumulator)\n --index\n }\n return
accumulator}\n\n/**\n * Performs the given [action] on each character.\n *\npublic inline fun
CharSequence.forEach(action: (Char) -> Unit): Unit {\n for (element in this) action(element)\n}\n\n/**\n *
Performs the given [action] on each character, providing sequential index with the character.\n * @param [action]
function that takes the index of a character and the character itself\n * and performs the desired action on the
character.\n *\npublic inline fun CharSequence.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {\n var
index = 0\n for (item in this) action(index++, item)\n}\n\n/**\n * Returns the largest character or `null` if there are
no characters.\n *\npublic fun CharSequence.max(): Char? {\n if (isEmpty()) return null\n var max = this[0]\n
for (i in 1..lastIndex) {\n val e = this[i]\n if (max < e) max = e\n }\n return max}\n\n/**\n * Returns
the first character yielding the largest value of the given function or `null` if there are no characters.\n * \n *
@sample samples.collections.Collections.Aggregates.maxBy\n *\npublic inline fun <R : Comparable<R>>
CharSequence.maxBy(selector: (Char) -> R): Char? {\n if (isEmpty()) return null\n var maxElem = this[0]\n
val lastIndex = this.lastIndex\n if (lastIndex == 0) return maxElem\n var maxValue = selector(maxElem)\n for
(i in 1..lastIndex) {\n val e = this[i]\n val v = selector(e)\n if (maxValue < v) {\n maxElem = e\n

```



```

    maxValue = v\n    }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first character having the largest
value according to the provided [comparator] or `null` if there are no characters.\n */\npublic fun
CharSequence.maxWith(comparator: Comparator<in Char>): Char? {\n    if (isEmpty()) return null\n    var max =
this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n
return max\n}\n\n/**\n * Returns the smallest character or `null` if there are no characters.\n */\npublic fun
CharSequence.min(): Char? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n
val e = this[i]\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the first character yielding the
smallest value of the given function or `null` if there are no characters.\n * \n * @sample
samples.collections.Collections.Aggregates.minBy\n */\npublic inline fun <R : Comparable<R>>
CharSequence.minBy(selector: (Char) -> R): Char? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val
lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in
1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n
minValue = v\n        }\n    }\n    return minElem\n}\n\n/**\n * Returns the first character having the smallest value
according to the provided [comparator] or `null` if there are no characters.\n */\npublic fun
CharSequence.minWith(comparator: Comparator<in Char>): Char? {\n    if (isEmpty()) return null\n    var min =
this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n
return min\n}\n\n/**\n * Returns `true` if the char sequence has no characters.\n * \n * @sample
samples.collections.Collections.Aggregates.none\n */\npublic fun CharSequence.none(): Boolean {\n    return
isEmpty()\n}\n\n/**\n * Returns `true` if no characters match the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.noneWithPredicate\n */\npublic inline fun
CharSequence.none(predicate: (Char) -> Boolean): Boolean {\n    for (element in this) if (predicate(element)) return
false\n    return true\n}\n\n/**\n * Performs the given [action] on each character and returns the char sequence itself
afterwards.\n */\n@SinceKotlin("1.1")\npublic inline fun <S : CharSequence> S.onEach(action: (Char) -> Unit): S
{\n    return apply { for (element in this) action(element) }\n}\n\n/**\n * Accumulates value starting with the first
character and applying [operation] from left to right to current accumulator value and each character.\n */\npublic
inline fun CharSequence.reduce(operation: (acc: Char, Char) -> Char): Char {\n    if (isEmpty())\n        throw
UnsupportedOperationException("Empty char sequence can't be reduced.")\n    var accumulator = this[0]\n    for
(index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return
accumulator\n}\n\n/**\n * Accumulates value starting with the first character and applying [operation] from left to
right\n * to current accumulator value and each character with its index in the original char sequence.\n * @param
[operation] function that takes the index of a character, current accumulator value\n * and the character itself and
calculates the next accumulator value.\n */\npublic inline fun CharSequence.reduceIndexed(operation: (index: Int,
acc: Char, Char) -> Char): Char {\n    if (isEmpty())\n        throw UnsupportedOperationException("Empty char
sequence can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator =
operation(index, accumulator, this[index])\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting
with last character and applying [operation] from right to left to each character and current accumulator value.\n */\n
public inline fun CharSequence.reduceRight(operation: (Char, acc: Char) -> Char): Char {\n    var index =
lastIndex\n    if (index < 0) throw UnsupportedOperationException("Empty char sequence can't be reduced.")\n    var
accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(get(index--), accumulator)\n
    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with last character and applying [operation] from
right to left\n * to each character with its index in the original char sequence and current accumulator value.\n *
@param [operation] function that takes the index of a character, the character itself\n * and current accumulator
value, and calculates the next accumulator value.\n */\npublic inline fun
CharSequence.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {\n    var index =
lastIndex\n    if (index < 0) throw UnsupportedOperationException("Empty char sequence can't be reduced.")\n    var
accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index),
accumulator)\n        --index\n    }\n    return accumulator\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each character in the char sequence.\n */\npublic inline fun

```

```

CharSequence.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

```

* Returns the sum of all values produced by [selector] function applied to each character in the char sequence.

```

public inline fun CharSequence.sumByDouble(selector: (Char) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

```

* Splits this char sequence into a list of strings each not exceeding the given [size].

* The last string in the resulting list may have less characters than the given [size].

* @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.

* @sample samples.collections.Collections.Transformations.chunked

```

@SinceKotlin("1.2")
public fun CharSequence.chunked(size: Int): List<String> {
    return windowed(size, size, partialWindows = true)
}

```

* Splits this char sequence into several char sequences each not exceeding the given [size] and applies the given [transform] function to an each.

* @return list of results of the [transform] applied to an each char sequence.

* Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function.

* You should not store it or allow it to escape in some way, unless you made a snapshot of it.

* The last char sequence may have less characters than the given [size].

* @param size the number of elements to take in each char sequence, must be positive and can be greater than the number of elements in this char sequence.

* @sample samples.text.Strings.chunkedTransform

```

@SinceKotlin("1.2")
public fun <R> CharSequence.chunked(size: Int, transform: (CharSequence) -> R): List<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

```

* Splits this char sequence into a sequence of strings each not exceeding the given [size].

* The last string in the resulting sequence may have less characters than the given [size].

* @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.

* @sample samples.collections.Collections.Transformations.chunked

```

@SinceKotlin("1.2")
public fun CharSequence.chunkedSequence(size: Int): Sequence<String> {
    return chunkedSequence(size) { it.toString() }
}

```

* Splits this char sequence into several char sequences each not exceeding the given [size] and applies the given [transform] function to an each.

* @return sequence of results of the [transform] applied to an each char sequence.

* Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function.

* You should not store it or allow it to escape in some way, unless you made a snapshot of it.

* The last char sequence may have less characters than the given [size].

* @param size the number of elements to take in each char sequence, must be positive and can be greater than the number of elements in this char sequence.

* @sample samples.text.Strings.chunkedTransformToSequence

```

@SinceKotlin("1.2")
public fun <R> CharSequence.chunkedSequence(size: Int, transform: (CharSequence) -> R): Sequence<R> {
    return windowedSequence(size, size, partialWindows = true, transform = transform)
}

```

* Splits the original char sequence into pair of char sequences, where *first* char sequence contains characters for which [predicate] yielded `true`, while *second* char sequence contains characters for which [predicate] yielded `false`.

```

public inline fun CharSequence.partition(predicate: (Char) -> Boolean): Pair<CharSequence, CharSequence> {
    val first = StringBuilder()
    val second = StringBuilder()
    for (element in this) {
        if (predicate(element)) {
            first.append(element)
        } else {
            second.append(element)
        }
    }
    return Pair(first, second)
}

```

* Splits the original string into pair of strings, where *first* string contains characters for which [predicate] yielded `true`, while *second* string contains characters for which [predicate] yielded `false`.

```

public inline fun String.partition(predicate: (Char) -> Boolean): Pair<String, String> {
    val first = StringBuilder()
    val second = StringBuilder()
    for (element in this) {
        if (predicate(element)) {
            first.append(element)
        } else {
            second.append(element)
        }
    }
    return Pair(first.toString(), second.toString())
}

```

* Returns a list of snapshots of the window of the given [size] sliding along this char sequence with the given [step], where each snapshot is a string.

* Several last strings may have less characters than the given [size].

* Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.

* @param size the number of elements to take in each window

* @param step the number of elements to move the window forward by on an each step, by default 1

* @param partialWindows controls whether or not to keep

partial windows in the end if any, by default `false` which means partial windows won't be preserved

```

@sample samples.collections.Sequences.Transformations.takeWindows
SinceKotlin("1.2")
public fun CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): List<String> {
    return windowed(size, step, partialWindows) { it.toString() }
}

```

Returns a list of results of applying the given [transform] function to an each char sequence representing a view over the window of the given [size] sliding along this char sequence with the given [step]. Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function. You should not store it or allow it to escape in some way, unless you made a snapshot of it. Several last char sequences may have less characters than the given [size]. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence. @param size the number of elements to take in each window @param step the number of elements to move the window forward by on an each step, by default 1 @param partialWindows controls whether or not to keep partial windows in the end if any, by default `false` which means partial windows won't be preserved

```

@sample samples.collections.Sequences.Transformations.averageWindows
SinceKotlin("1.2")
public fun <R> CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (CharSequence) -> R): List<R> {
    checkWindowSizeStep(size, step)
    val thisSize = this.length
    val result = ArrayList<R>((thisSize + step - 1) / step)
    var index = 0
    while (index < thisSize) {
        val end = index + size
        val coercedEnd = if (end > thisSize) { if (partialWindows) thisSize else break }
        else end
        result.add(transform(subSequence(index, coercedEnd)))
        index += step
    }
    return result
}

```

Returns a sequence of snapshots of the window of the given [size] sliding along this char sequence with the given [step], where each snapshot is a string. Several last strings may have less characters than the given [size]. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence. @param size the number of elements to take in each window @param step the number of elements to move the window forward by on an each step, by default 1 @param partialWindows controls whether or not to keep partial windows in the end if any, by default `false` which means partial windows won't be preserved

```

@sample samples.collections.Sequences.Transformations.takeWindows
SinceKotlin("1.2")
public fun CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false): Sequence<String> {
    return windowedSequence(size, step, partialWindows) { it.toString() }
}

```

Returns a sequence of results of applying the given [transform] function to an each char sequence representing a view over the window of the given [size] sliding along this char sequence with the given [step]. Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function. You should not store it or allow it to escape in some way, unless you made a snapshot of it. Several last char sequences may have less characters than the given [size]. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence. @param size the number of elements to take in each window @param step the number of elements to move the window forward by on an each step, by default 1 @param partialWindows controls whether or not to keep partial windows in the end if any, by default `false` which means partial windows won't be preserved

```

@sample samples.collections.Sequences.Transformations.averageWindows
SinceKotlin("1.2")
public fun <R> CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (CharSequence) -> R): Sequence<R> {
    checkWindowSizeStep(size, step)
    val windows = (if (partialWindows) indices else 0 until length - size + 1) step step
    return windows.asSequence().map { index -> transform(subSequence(index, (index + size).coerceAtMost(length))) }
}

```

Returns a list of pairs built from the characters of `this` and the [other] char sequences with the same index. The returned list has length of the shortest char sequence. @sample samples.text.Strings.zip

```

public infix fun CharSequence.zip(other: CharSequence): List<Pair<Char, Char>> {
    return zip(other) { c1, c2 -> c1 to c2 }
}

```

Returns a list of values built from the characters of `this` and the [other] char sequences with the same index using the provided [transform] function applied to each pair of characters. The returned list has length of the shortest char sequence. @sample samples.text.Strings.zipWithTransform

```

public inline fun <V> CharSequence.zip(other: CharSequence, transform: (a: Char, b: Char) -> V): List<V> {
    val length =

```

```

minOf(this.length, other.length)\n  val list = ArrayList<V>(length)\n  for (i in 0 until length) {\n
list.add(transform(this[i], other[i]))\n  }\n  return list\n}\n\n/**\n * Returns a list of pairs of each two adjacent
characters in this char sequence.\n * \n * The returned list is empty if this char sequence contains less than two
characters.\n * \n * @sample samples.collections.Collections.Transformations.zipWithNext\n
*\n\n@SinceKotlin("1.2")\npublic fun CharSequence.zipWithNext(): List<Pair<Char, Char>> {\n  return
zipWithNext { a, b -> a to b }\n}\n\n/**\n * Returns a list containing the results of applying the given [transform]
function\n * to an each pair of two adjacent characters in this char sequence.\n * \n * The returned list is empty if
this char sequence contains less than two characters.\n * \n * @sample
samples.collections.Collections.Transformations.zipWithNextToFindDeltas\n
*\n\n@SinceKotlin("1.2")\npublic
inline fun <R> CharSequence.zipWithNext(transform: (a: Char, b: Char) -> R): List<R> {\n  val size = length - 1\n
if (size < 1) return emptyList()\n  val result = ArrayList<R>(size)\n  for (index in 0 until size) {\n
result.add(transform(this[index], this[index + 1]))\n  }\n  return result\n}\n\n/**\n * Creates an [Iterable] instance
that wraps the original char sequence returning its characters when being iterated.\n *\n\npublic fun
CharSequence.asIterable(): Iterable<Char> {\n  if (this is String && isEmpty()) return emptyList()\n  return
Iterable { this.iterator() }\n}\n\n/**\n * Creates a [Sequence] instance that wraps the original char sequence
returning its characters when being iterated.\n *\n\npublic fun CharSequence.asSequence(): Sequence<Char> {\n  if
(this is String && isEmpty()) return emptySequence()\n  return Sequence { this.iterator() }\n}\n\n"/*\n
Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code
is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*\n\n\n@file:kotlin.jvm.JvmMultifileClass\n\n@file:kotlin.jvm.JvmName("StringsKt")\n\n\npackage
kotlin.text\n\nimport kotlin.contracts.contract\n\n\n/**\n * Returns a sub sequence of this char sequence
having leading and trailing characters matching the [predicate] removed.\n *\n\n\npublic inline fun
CharSequence.trim(predicate: (Char) -> Boolean): CharSequence {\n  var startIndex = 0\n  var endIndex =
length - 1\n  var startFound = false\n  while (startIndex <= endIndex) {\n    val index = if (!startFound)
startIndex else endIndex\n    val match = predicate(this[index])\n    if (!startFound) {\n      if
(!match)\n        startFound = true\n      else\n        startIndex += 1\n    } else {\n      if
(!match)\n        break\n      else\n        endIndex -= 1\n    }\n  }\n  return
subSequence(startIndex, endIndex + 1)\n}\n\n/**\n * Returns a string having leading and trailing characters
matching the [predicate] removed.\n *\n\n\npublic inline fun String.trim(predicate: (Char) -> Boolean): String =\n
(this as CharSequence).trim(predicate).toString()\n}\n\n/**\n * Returns a sub sequence of this char sequence
having leading characters matching the [predicate] removed.\n *\n\n\npublic inline fun
CharSequence.trimStart(predicate: (Char) -> Boolean): CharSequence {\n  for (index in this.indices)\n    if
(!predicate(this[index]))\n      return subSequence(index, length)\n  return ""\n}\n\n/**\n * Returns a string
having leading characters matching the [predicate] removed.\n *\n\n\npublic inline fun
String.trimStart(predicate: (Char) -> Boolean): String =\n (this as
CharSequence).trimStart(predicate).toString()\n}\n\n/**\n * Returns a sub sequence of this char sequence having
trailing characters matching the [predicate] removed.\n *\n\n\npublic inline fun CharSequence.trimEnd(predicate:
(Char) -> Boolean): CharSequence {\n  for (index in this.indices.reversed())\n    if
(!predicate(this[index]))\n      return subSequence(0, index + 1)\n  return ""\n}\n\n/**\n * Returns
a string having trailing characters matching the [predicate] removed.\n *\n\n\npublic inline fun
String.trimEnd(predicate: (Char) -> Boolean): String =\n (this as
CharSequence).trimEnd(predicate).toString()\n}\n\n/**\n * Returns a sub sequence of this char sequence having
leading and trailing characters from the [chars] array removed.\n *\n\n\npublic fun CharSequence.trim(vararg chars:
Char): CharSequence = trim { it in chars }\n}\n\n/**\n * Returns a string having leading and trailing characters
from the [chars] array removed.\n *\n\n\npublic fun String.trim(vararg chars: Char): String = trim { it in chars
}\n}\n\n/**\n * Returns a sub sequence of this char sequence having leading characters from the [chars] array
removed.\n *\n\n\npublic fun CharSequence.trimStart(vararg chars: Char): CharSequence = trimStart { it in chars
}\n}\n\n/**\n * Returns a string having leading characters from the [chars] array removed.\n *\n\n\npublic fun

```

```

String.trimStart(vararg chars: Char): String = trimStart { it in chars }
CharSequence.trimEnd(vararg chars: Char): CharSequence = trimEnd { it in chars }
String = trimEnd { it in chars }
String = trimEnd { it in chars }
CharSequence.trim(): CharSequence = trim(Char::isWhitespace)
String = trim(Char::isWhitespace)
String.trim(): String = (this as CharSequence).trim().toString()
CharSequence.trimStart(): CharSequence = trimStart(Char::isWhitespace)
String.trimStart(): String = (this as CharSequence).trimStart().toString()
CharSequence.trimEnd(): CharSequence = trimEnd(Char::isWhitespace)
String.trimEnd(): String = (this as CharSequence).trimEnd().toString()
String.padStart(length: Int, padChar: Char = ' '): CharSequence {
    if (length < 0) throw IllegalArgumentException("Desired length $length is less than zero.")
    if (length <= this.length) return this.subSequence(0, this.length)
    val sb = StringBuilder(length)
    for (i in 1..(length - this.length)) sb.append(padChar)
    sb.append(this)
    return sb
}
String.padStart(length: Int, padChar: Char = ' '): String = (this as CharSequence).padStart(length, padChar).toString()
CharSequence.padEnd(length: Int, padChar: Char = ' '): CharSequence {
    if (length < 0) throw IllegalArgumentException("Desired length $length is less than zero.")
    if (length <= this.length) return this.subSequence(0, this.length)
    val sb = StringBuilder(length)
    sb.append(this)
    for (i in 1..(length - this.length)) sb.append(padChar)
    return sb
}
String.padEnd(length: Int, padChar: Char = ' '): String = (this as CharSequence).padEnd(length, padChar).toString()
String.isNullOrEmpty(): Boolean {
    contract
}

```

```

{
    returns(false) implies (this@isEmpty != null)
}
return this == null || this.length ==
0
}
Returns `true` if this char sequence is empty (contains no characters).
@sample
samples.text.Strings.stringIsEmpty
@kotlin.internal.InlineOnly
public inline fun
CharSequence.isEmpty(): Boolean = length == 0
Returns `true` if this char sequence is not
empty.
@sample
samples.text.Strings.stringIsNotEmpty
@kotlin.internal.InlineOnly
public
inline fun CharSequence.isNotEmpty(): Boolean = length > 0
implemented differently in JVM and
JS
/public fun String.isBlank(): Boolean = length() == 0 || all { it.isWhitespace() }
Returns
`true` if this char sequence is not empty and contains some characters except of whitespace characters.
@sample
samples.text.Strings.stringIsNotBlank
@kotlin.internal.InlineOnly
public inline fun
CharSequence.isNotBlank(): Boolean = !isBlank()
Returns `true` if this nullable char sequence is
either `null` or empty or consists solely of whitespace characters.
@sample
samples.text.Strings.stringOrNullBlank
@kotlin.internal.InlineOnly
public inline fun
CharSequence?.isNullOrBlank(): Boolean {
    contract {
        returns(false) implies (this@isNullOrBlank !=
null)
}
return this == null || this.isBlank()
}
Iterator for characters of the given char
sequence.
public operator fun CharSequence.iterator(): CharIterator = object : CharIterator() {
    private
var index = 0
    public override fun nextChar(): Char = get(index++)
    public override fun hasNext():
Boolean = index < length
}
Returns the string if it is not `null`, or the empty string otherwise.
@kotlin.internal.InlineOnly
public inline fun String?.orEmpty(): String = this ?: ""
Returns
this char sequence if it's not empty or the result of calling [defaultValue] function if the char sequence is
empty.
@sample
samples.text.Strings.stringIfEmpty
@kotlin.SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifEmpty(defaultValue: () -
> R): R where C : CharSequence, C : R =
    if (isEmpty()) defaultValue() else this
Returns this
char sequence if it is not empty and doesn't consist solely of whitespace characters, or the result of calling
[defaultValue] function otherwise.
@sample
samples.text.Strings.stringIfBlank
@kotlin.SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifBlank(defaultValue: () -
> R): R where C : CharSequence, C : R =
    if (isBlank()) defaultValue() else this
Returns the
range of valid character indices for this char sequence.
public val CharSequence.indices: IntRange
get() = 0..length - 1
Returns the index of the last character in the char sequence or -1 if it is
empty.
public val CharSequence.lastIndex: Int
get() = this.length - 1
Returns `true` if
this CharSequence has Unicode surrogate pair at the specified [index].
public fun
CharSequence.hasSurrogatePairAt(index: Int): Boolean {
    return index in 0..length - 2
    &&
this[index].isHighSurrogate()
    && this[index + 1].isLowSurrogate()
}
Returns a
substring specified by the given [range] of indices.
public fun String.substring(range: IntRange): String =
substring(range.start, range.endInclusive + 1)
Returns a subsequence of this char sequence specified
by the given [range] of indices.
public fun CharSequence.subSequence(range: IntRange): CharSequence =
subSequence(range.start, range.endInclusive + 1)
Returns a subsequence of this char sequence.
This extension is chosen only for invocation with old-named parameters.
Replace parameter names with
the same as those of [CharSequence.subSequence].
@kotlin.internal.InlineOnly
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false
warning
@Deprecated("Use parameters named startIndex and endIndex.",
ReplaceWith("subSequence(startIndex = start, endIndex = end)"))
public inline fun String.subSequence(start:
Int, end: Int): CharSequence = subSequence(start, end)
Returns a substring of chars from a range of
this char sequence starting at the [startIndex] and ending right before the [endIndex].
@param startIndex
the start index (inclusive).
@param endIndex the end index (exclusive). If not specified, the length of the char
sequence is used.
@kotlin.internal.InlineOnly
public inline fun CharSequence.substring(startIndex: Int,
endIndex: Int = length): String = subSequence(startIndex, endIndex).toString()
Returns a substring of
chars at indices from the specified [range] of this char sequence.
public fun CharSequence.substring(range:
IntRange): String = subSequence(range.start, range.endInclusive + 1).toString()
Returns a substring

```

before the first occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
`public fun String.substringBefore(delimiter: Char, missingDelimiterValue: String = this): String {`
 `val index = indexOf(delimiter)`
 `return if (index == -1) missingDelimiterValue else substring(0, index)`
 * Returns a substring before the first occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
`public fun String.substringBefore(delimiter: String, missingDelimiterValue: String = this): String {`
 `val index = indexOf(delimiter)`
 `return if (index == -1) missingDelimiterValue else substring(0, index)`
 * Returns a substring after the first occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
`public fun String.substringAfter(delimiter: Char, missingDelimiterValue: String = this): String {`
 `val index = indexOf(delimiter)`
 `return if (index == -1) missingDelimiterValue else substring(index + 1, length)`
 * Returns a substring after the first occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
`public fun String.substringAfter(delimiter: String, missingDelimiterValue: String = this): String {`
 `val index = indexOf(delimiter)`
 `return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)`
 * Returns a substring before the last occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
`public fun String.substringBeforeLast(delimiter: Char, missingDelimiterValue: String = this): String {`
 `val index = lastIndexOf(delimiter)`
 `return if (index == -1) missingDelimiterValue else substring(0, index)`
 * Returns a substring before the last occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
`public fun String.substringBeforeLast(delimiter: String, missingDelimiterValue: String = this): String {`
 `val index = lastIndexOf(delimiter)`
 `return if (index == -1) missingDelimiterValue else substring(0, index)`
 * Returns a substring after the last occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
`public fun String.substringAfterLast(delimiter: Char, missingDelimiterValue: String = this): String {`
 `val index = lastIndexOf(delimiter)`
 `return if (index == -1) missingDelimiterValue else substring(index + 1, length)`
 * Returns a substring after the last occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
`public fun String.substringAfterLast(delimiter: String, missingDelimiterValue: String = this): String {`
 `val index = lastIndexOf(delimiter)`
 `return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)`
 * Returns a char sequence with content of this char sequence where its part at the given range is replaced with the [replacement] char sequence.
 * @param startIndex the index of the first character to be replaced.
 * @param endIndex the index of the first character after the replacement to keep in the string.
`public fun CharSequence.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): CharSequence {`
 `if (endIndex < startIndex) throw IndexOutOfBoundsException("End index ($endIndex) is less than start index ($startIndex).")`
 `val sb = StringBuilder()`
 `sb.append(this, 0, startIndex)`
 `sb.append(replacement)`
 `sb.append(this, endIndex, length)`
 `return sb`
 * Replaces the part of the string at the given range with the [replacement] char sequence.
 * @param startIndex the index of the first character to be replaced.
 * @param endIndex the index of the first character after the replacement to keep in the string.
`@kotlin.internal.InlineOnly`
`public inline fun String.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): String =`
 `(this as CharSequence).replaceRange(startIndex, endIndex, replacement).toString()`
 * Returns a char sequence with content of this char sequence where its part at the given [range] is replaced with the [replacement] char sequence.
 * The end index of the [range] is included in the part to be replaced.
`public fun CharSequence.replaceRange(range: IntRange, replacement: CharSequence): CharSequence =`
 `replaceRange(range.start, range.endInclusive + 1, replacement)`
 * Replace the part of string at the given [range] with the [replacement] string.
 * The end index of the [range] is included in the part to be

```

replaced.\r\n *@\r\n@kotlin.internal.InlineOnly\r\npublic inline fun String.replaceRange(range: IntRange,
replacement: CharSequence): String =\r\n (this as CharSequence).replaceRange(range,
replacement).toString()\r\n\r\n/**\r\n * Returns a char sequence with content of this char sequence where its part at
the given range is removed.\r\n * @param startIndex the index of the first character to be removed.\r\n *
@param endIndex the index of the first character after the removed part to keep in the string.\r\n * [endIndex]
is not included in the removed part.\r\n */\r\npublic fun CharSequence.removeRange(startIndex: Int, endIndex: Int):
CharSequence {\r\n if (endIndex < startIndex)\r\n throw IndexOutOfBoundsException("\r\nEnd index
($endIndex) is less than start index ($startIndex).")\r\n\r\n if (endIndex == startIndex)\r\n return
this.subSequence(0, length)\r\n\r\n val sb = StringBuilder(length - (endIndex - startIndex))\r\n sb.append(this, 0,
startIndex)\r\n sb.append(this, endIndex, length)\r\n return sb\r\n}\r\n\r\n/**\r\n * Removes the part of a string at
a given range.\r\n * @param startIndex the index of the first character to be removed.\r\n * @param endIndex the
index of the first character after the removed part to keep in the string.\r\n * [endIndex] is not included in the
removed part.\r\n */\r\n@kotlin.internal.InlineOnly\r\npublic inline fun String.removeRange(startIndex: Int,
endIndex: Int): String =\r\n (this as CharSequence).removeRange(startIndex, endIndex).toString()\r\n\r\n/**\r\n * Returns a char sequence with content of this char sequence where its part at the given [range] is removed.\r\n * The end index of the [range] is included in the removed part.\r\n */\r\npublic fun
CharSequence.removeRange(range: IntRange): CharSequence = removeRange(range.start, range.endInclusive +
1)\r\n\r\n/**\r\n * Removes the part of a string at the given [range].\r\n * The end index of the [range] is
included in the removed part.\r\n */\r\n@kotlin.internal.InlineOnly\r\npublic inline fun String.removeRange(range:
IntRange): String =\r\n (this as CharSequence).removeRange(range).toString()\r\n\r\n/**\r\n * If this char
sequence starts with the given [prefix], returns a new char sequence\r\n * with the prefix removed. Otherwise,
returns a new char sequence with the same characters.\r\n */\r\npublic fun CharSequence.removePrefix(prefix:
CharSequence): CharSequence {\r\n if (startsWith(prefix)) {\r\n return subSequence(prefix.length, length)\r\n
}\r\n return subSequence(0, length)\r\n}\r\n\r\n/**\r\n * If this string starts with the given [prefix], returns a copy
of this string\r\n * with the prefix removed. Otherwise, returns this string.\r\n */\r\npublic fun
String.removePrefix(prefix: CharSequence): String {\r\n if (startsWith(prefix)) {\r\n return
substring(prefix.length)\r\n }\r\n return this\r\n}\r\n\r\n/**\r\n * If this char sequence ends with the given
[suffix], returns a new char sequence\r\n * with the suffix removed. Otherwise, returns a new char sequence with the
same characters.\r\n */\r\npublic fun CharSequence.removeSuffix(suffix: CharSequence): CharSequence {\r\n if
(endsWith(suffix)) {\r\n return subSequence(0, length - suffix.length)\r\n }\r\n return subSequence(0,
length)\r\n}\r\n\r\n/**\r\n * If this string ends with the given [suffix], returns a copy of this string\r\n * with the
suffix removed. Otherwise, returns this string.\r\n */\r\npublic fun String.removeSuffix(suffix: CharSequence):
String {\r\n if (endsWith(suffix)) {\r\n return substring(0, length - suffix.length)\r\n }\r\n return
this\r\n}\r\n\r\n/**\r\n * When this char sequence starts with the given [prefix] and ends with the given [suffix],\r\n *
returns a new char sequence having both the given [prefix] and [suffix] removed.\r\n * Otherwise returns a new char
sequence with the same characters.\r\n */\r\npublic fun CharSequence.removeSurrounding(prefix: CharSequence,
suffix: CharSequence): CharSequence {\r\n if ((length >= prefix.length + suffix.length) && startsWith(prefix) &&
endsWith(suffix)) {\r\n return subSequence(prefix.length, length - suffix.length)\r\n }\r\n return
subSequence(0, length)\r\n}\r\n\r\n/**\r\n * Removes from a string both the given [prefix] and [suffix] if and only
if\r\n * it starts with the [prefix] and ends with the [suffix].\r\n * Otherwise returns this string unchanged.\r\n
*/\r\npublic fun String.removeSurrounding(prefix: CharSequence, suffix: CharSequence): String {\r\n if ((length
>= prefix.length + suffix.length) && startsWith(prefix) && endsWith(suffix)) {\r\n return
substring(prefix.length, length - suffix.length)\r\n }\r\n return this\r\n}\r\n\r\n/**\r\n * When this char sequence
starts with and ends with the given [delimiter],\r\n * returns a new char sequence having this [delimiter] removed
both from the start and end.\r\n * Otherwise returns a new char sequence with the same characters.\r\n */\r\npublic
fun CharSequence.removeSurrounding(delimiter: CharSequence): CharSequence = removeSurrounding(delimiter,
delimiter)\r\n\r\n/**\r\n * Removes the given [delimiter] string from both the start and the end of this string\r\n * if
and only if it starts with and ends with the [delimiter].\r\n * Otherwise returns this string unchanged.\r\n */\r\npublic

```



```

sequence, \r\n * starting from the specified [startIndex] and optionally ignoring the case. \r\n * \r\n * @param
ignoreCase `true` to ignore character case when matching a character. By default `false`. \r\n * @return An index of
the first occurrence of matched character from [chars] or -1 if none of [chars] are found. \r\n * \r\n * \r\n\npublic fun
CharSequence.indexOfAny(chars: CharArray, startIndex: Int = 0, ignoreCase: Boolean = false): Int { \r\n    if
(!ignoreCase && chars.size == 1 && this is String) { \r\n        val char = chars.single() \r\n        return
nativeIndexOf(char, startIndex) \r\n    } \r\n    for (index in startIndex.coerceAtLeast(0)..lastIndex) { \r\n        val
charAtIndex = get(index) \r\n        if (chars.any { it.equals(charAtIndex, ignoreCase) }) \r\n            return index \r\n
    } \r\n    return -1 \r\n} \r\n\n\r\n\n** \r\n * Finds the index of the last occurrence of any of the specified [chars] in this char
sequence, \r\n * starting from the specified [startIndex] and optionally ignoring the case. \r\n * \r\n * @param
startIndex The index of character to start searching at. The search proceeds backward toward the beginning of the
string. \r\n * @param ignoreCase `true` to ignore character case when matching a character. By default `false`. \r\n *
\r\n * @return An index of the last occurrence of matched character from [chars] or -1 if none of [chars] are found. \r\n
\r\n * \r\n\npublic fun CharSequence.lastIndexOfAny(chars: CharArray, startIndex: Int = lastIndex, ignoreCase:
Boolean = false): Int { \r\n    if (!ignoreCase && chars.size == 1 && this is String) { \r\n        val char =
chars.single() \r\n        return nativeLastIndexOf(char, startIndex) \r\n    } \r\n    for (index in
startIndex.coerceAtMost(lastIndex) downTo 0) { \r\n        val charAtIndex = get(index) \r\n        if (chars.any {
it.equals(charAtIndex, ignoreCase) }) \r\n            return index \r\n    } \r\n    return -1 \r\n} \r\n\n\r\n\nprivate fun
CharSequence.indexOf(other: CharSequence, startIndex: Int, endIndex: Int, ignoreCase: Boolean, last: Boolean =
false): Int { \r\n    val indices = if (!last) \r\n        startIndex.coerceAtLeast(0)..endIndex.coerceAtMost(length) \r\n
    else \r\n        startIndex.coerceAtMost(lastIndex) downTo endIndex.coerceAtLeast(0) \r\n    if (this is String &&
other is String) { // smart cast \r\n        for (index in indices) { \r\n            if (other.regionMatches(0, this, index,
other.length, ignoreCase)) \r\n                return index \r\n        } \r\n    } else { \r\n        for (index in indices) { \r\n
if (other.regionMatchesImpl(0, this, index, other.length, ignoreCase)) \r\n            return index \r\n        } \r\n
    } \r\n    return -1 \r\n} \r\n\n\r\nprivate fun CharSequence.findAnyOf(strings: Collection<String>, startIndex: Int, ignoreCase:
Boolean, last: Boolean): Pair<Int, String>? { \r\n    if (!ignoreCase && strings.size == 1) { \r\n        val string =
strings.single() \r\n        val index = if (!last) indexOf(string, startIndex) else lastIndexOf(string, startIndex) \r\n
return if (index < 0) null else index to string \r\n    } \r\n    val indices = if (!last)
startIndex.coerceAtLeast(0)..length else startIndex.coerceAtMost(lastIndex) downTo 0 \r\n    if (this is String)
{ \r\n        for (index in indices) { \r\n            val matchingString = strings.firstOrNull { it.regionMatches(0, this,
index, it.length, ignoreCase) } \r\n            if (matchingString != null) \r\n                return index to matchingString \r\n
        } \r\n    } else { \r\n        for (index in indices) { \r\n            val matchingString = strings.firstOrNull {
it.regionMatchesImpl(0, this, index, it.length, ignoreCase) } \r\n            if (matchingString != null) \r\n
return index to matchingString \r\n        } \r\n    } \r\n    return null \r\n} \r\n\n\r\n\n** \r\n * Finds the first occurrence of
any of the specified [strings] in this char sequence, \r\n * starting from the specified [startIndex] and optionally
ignoring the case. \r\n * \r\n * @param ignoreCase `true` to ignore character case when matching a string. By default
`false`. \r\n * @return A pair of an index of the first occurrence of matched string from [strings] and the string
matched \r\n * or `null` if none of [strings] are found. \r\n * \r\n * To avoid ambiguous results when strings in [strings]
have characters in common, this method proceeds from \r\n * the beginning to the end of this string, and finds at each
position the first element in [strings] \r\n * that matches this string at that position. \r\n * \r\n\npublic fun
CharSequence.findAnyOf(strings: Collection<String>, startIndex: Int = 0, ignoreCase: Boolean = false): Pair<Int,
String>? = \r\n    findAnyOf(strings, startIndex, ignoreCase, last = false) \r\n\n\r\n\n** \r\n * Finds the last occurrence of
any of the specified [strings] in this char sequence, \r\n * starting from the specified [startIndex] and optionally
ignoring the case. \r\n * \r\n * @param startIndex The index of character to start searching at. The search proceeds
backward toward the beginning of the string. \r\n * @param ignoreCase `true` to ignore character case when
matching a string. By default `false`. \r\n * @return A pair of an index of the last occurrence of matched string from
[strings] and the string matched or `null` if none of [strings] are found. \r\n * \r\n * To avoid ambiguous results when
strings in [strings] have characters in common, this method proceeds from \r\n * the end toward the beginning of this
string, and finds at each position the first element in [strings] \r\n * that matches this string at that position. \r\n

```

```

public fun CharSequence.findLastAnyOf(strings: Collection<String>, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Pair<Int, String>? = findAnyOf(strings, startIndex, ignoreCase, last = true)
public fun CharSequence.findAnyOf(strings: Collection<String>, startIndex: Int = 0, ignoreCase: Boolean = false): Int = findAnyOf(strings, startIndex, ignoreCase, last = false).first ?: -1
public fun CharSequence.indexOfAny(strings: Collection<String>, startIndex: Int = 0, ignoreCase: Boolean = false): Int = findAnyOf(strings, startIndex, ignoreCase, last = true).first ?: -1
public fun CharSequence.indexOfAny(char: Char, startIndex: Int = 0, ignoreCase: Boolean = false): Int = return if (ignoreCase || this !is String) indexOfAny(charArrayOf(char), startIndex, ignoreCase) else nativeIndexOf(char, startIndex)
public fun CharSequence.indexOf(string: String, startIndex: Int = 0, ignoreCase: Boolean = false): Int = return if (ignoreCase || this !is String) indexOf(string, startIndex, length, ignoreCase) else nativeIndexOf(string, startIndex)
public fun CharSequence.indexOf(char: Char, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int = return if (ignoreCase || this !is String) lastIndexOfAny(charArrayOf(char), startIndex, ignoreCase) else nativeLastIndexOf(char, startIndex)
public fun CharSequence.indexOf(string: String, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int = return if (ignoreCase || this !is String) indexOf(string, startIndex, 0, ignoreCase, last = true) else nativeLastIndexOf(string, startIndex)
public fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean = if (other is String)

```



```

include terminating line separators.\r\n */\r\npublic fun CharSequence.lineSequence(): Sequence<String> =
splitToSequence("\\r\\n", "\\n", "\\r")\r\n\r\n/**\r\n * Splits this char sequence to a list of lines delimited by any
of the following character sequences: CRLF, LF or CR.\r\n * The lines returned do not include terminating line
separators.\r\n */\r\npublic fun CharSequence.lines(): List<String> = lineSequence().toList()\r\n", /*\r\n * Copyright
2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is
governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage
kotlin.text\r\n\r\nimport kotlin.js.RegExp\r\n\r\n/**\r\n * Converts the characters in the specified array to a
string.\r\n */\r\n@SinceKotlin("1.2")\r\npublic actual fun String(chars: CharArray): String {\r\n    var result =
""\r\n    for (char in chars) {\r\n        result += char\r\n    }\r\n    return result\r\n}\r\n\r\n/**\r\n * Converts the
characters from a portion of the specified array to a string.\r\n * @throws IndexOutOfBoundsException if
either [offset] or [length] are less than zero\r\n * or `offset + length` is out of [chars] array bounds.\r\n
*/\r\n@SinceKotlin("1.2")\r\npublic actual fun String(chars: CharArray, offset: Int, length: Int): String {\r\n    if
(offset < 0 || length < 0 || chars.size - offset < length)\r\n        throw IndexOutOfBoundsException("size:
${chars.size}; offset: $offset; length: $length")\r\n    var result = ""\r\n    for (index in offset until offset + length)
{\r\n        result += chars[index]\r\n    }\r\n    return result\r\n}\r\n\r\n/**\r\n * Concatenates characters in this
[CharArray] into a String.\r\n */\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun
CharArray.concatToString(): String {\r\n    var result = ""\r\n    for (char in this) {\r\n        result += char\r\n    }\r\n
return result\r\n}\r\n\r\n/**\r\n * Concatenates characters in this [CharArray] or its subrange into a String.\r\n
*/\r\n * @param startIndex the beginning (inclusive) of the subrange of characters, 0 by default.\r\n * @param endIndex
the end (exclusive) of the subrange of characters, size of this array by default.\r\n * @throws
IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.\r\n
*/\r\n * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].\r\n\r\n
*/\r\n@SinceKotlin("1.3")\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\n@E
xperimentalStdlibApi\r\npublic actual fun CharArray.concatToString(startIndex: Int = 0, endIndex: Int = this.size):
String {\r\n    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)\r\n    var result = ""\r\n    for
(index in startIndex until endIndex) {\r\n        result += this[index]\r\n    }\r\n    return result\r\n}\r\n\r\n
Returns a [CharArray] containing characters of this string.\r\n\r\n
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun String.toCharArray(): CharArray {\r\n
return CharArray(length) { get(it) }\r\n}\r\n\r\n/**\r\n * Returns a [CharArray] containing characters of this string
or its substring.\r\n * @param startIndex the beginning (inclusive) of the substring, 0 by default.\r\n * @param
endIndex the end (exclusive) of the substring, length of this string by default.\r\n * @throws
IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the length of this
string.\r\n * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].\r\n\r\n
*/\r\n@SinceKotlin("1.3")\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\n@E
xperimentalStdlibApi\r\npublic actual fun String.toCharArray(startIndex: Int = 0, endIndex: Int = this.length):
CharArray {\r\n    AbstractList.checkBoundsIndexes(startIndex, endIndex, length)\r\n    return CharArray(endIndex
- startIndex) { get(startIndex + it) }\r\n}\r\n\r\n/**\r\n * Decodes a string from the bytes in UTF-8 encoding in this
array.\r\n * Malformed byte sequences are replaced by the replacement char `\\uFFFF`.\r\n\r\n
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun ByteArray.decodeToString(): String
{\r\n    return decodeUtf8(this, 0, size, false)\r\n}\r\n\r\n/**\r\n * Decodes a string from the bytes in UTF-8 encoding
in this array or its subrange.\r\n * @param startIndex the beginning (inclusive) of the subrange to decode, 0 by
default.\r\n * @param endIndex the end (exclusive) of the subrange to decode, size of this array by default.\r\n
*/\r\n * @param throwOnInvalidSequence specifies whether to throw an exception on malformed byte sequence or replace
it by the replacement char `\\uFFFF`.\r\n * @throws IndexOutOfBoundsException if [startIndex] is less than
zero or [endIndex] is greater than the size of this array.\r\n * @throws IllegalArgumentException if [startIndex] is
greater than [endIndex].\r\n * @throws CharacterCodingException if the byte array contains malformed UTF-8 byte
sequence and [throwOnInvalidSequence] is true.\r\n\r\n
*/\r\n@SinceKotlin("1.3")\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\n@E

```

```

experimentalStdlibApi\r\npublic actual fun ByteArray.decodeToString(\r\n  startIndex: Int = 0,\r\n  endIndex: Int =
this.size,\r\n  throwOnInvalidSequence: Boolean = false\r\n): String {\r\n
AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)\r\n  return decodeUtf8(this, startIndex, endIndex,
throwOnInvalidSequence)\r\n}\r\n\r\n/**\r\n * Encodes this string to an array of bytes in UTF-8 encoding.\r\n *\r\n
* Any malformed char sequence is replaced by the replacement byte sequence.\r\n
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun String.encodeToByteArray():
ByteArray {\r\n  return encodeUtf8(this, 0, length, false)\r\n}\r\n\r\n/**\r\n * Encodes this string or its substring to
an array of bytes in UTF-8 encoding.\r\n *\r\n * @param startIndex the beginning (inclusive) of the substring to
encode, 0 by default.\r\n * @param endIndex the end (exclusive) of the substring to encode, length of this string by
default.\r\n * @param throwOnInvalidSequence specifies whether to throw an exception on malformed char
sequence or replace.\r\n *\r\n * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex]
is greater than the length of this string.\r\n * @throws IllegalArgumentException if [startIndex] is greater than
[endIndex].\r\n * @throws CharacterCodingException if this string contains malformed char sequence and
[throwOnInvalidSequence] is true.\r\n
*/\r\n@SinceKotlin("1.3")\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\n@E
xperimentalStdlibApi\r\npublic actual fun String.encodeToByteArray(\r\n  startIndex: Int = 0,\r\n  endIndex: Int =
this.length,\r\n  throwOnInvalidSequence: Boolean = false\r\n): ByteArray {\r\n
AbstractList.checkBoundsIndexes(startIndex, endIndex, length)\r\n  return encodeUtf8(this, startIndex, endIndex,
throwOnInvalidSequence)\r\n}\r\n\r\n/**\r\n * Returns a copy of this string converted to upper case using the rules
of the default locale.\r\n *\r\n * @sample samples.text.Strings.toUpperCase\r\n
*/\r\n@kotlin.internal.InlineOnly\r\npublic actual inline fun String.toUpperCase(): String =
asDynamic().toUpperCase()\r\n\r\n/**\r\n * Returns a copy of this string converted to lower case using the rules of
the default locale.\r\n *\r\n * @sample samples.text.Strings.toLowerCase\r\n
*/\r\n@kotlin.internal.InlineOnly\r\npublic actual inline fun String.toLowerCase(): String =
asDynamic().toLowerCase()\r\n\r\n@kotlin.internal.InlineOnly\r\ninternal actual inline fun
String.nativeIndexOf(str: String, fromIndex: Int): Int = asDynamic().indexOf(str,
fromIndex)\r\n\r\n@kotlin.internal.InlineOnly\r\ninternal actual inline fun String.nativeLastIndexOf(str: String,
fromIndex: Int): Int = asDynamic().lastIndexOf(str, fromIndex)\r\n\r\n@kotlin.internal.InlineOnly\r\ninternal inline
fun String.nativeStartsWith(s: String, position: Int): Boolean = asDynamic().startsWith(s,
position)\r\n\r\n@kotlin.internal.InlineOnly\r\ninternal inline fun String.nativeEndsWith(s: String): Boolean =
asDynamic().endsWith(s)\r\n\r\n@kotlin.internal.InlineOnly\r\npublic actual inline fun String.substring(startIndex:
Int): String = asDynamic().substring(startIndex)\r\n\r\n@kotlin.internal.InlineOnly\r\npublic actual inline fun
String.substring(startIndex: Int, endIndex: Int): String = asDynamic().substring(startIndex,
endIndex)\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun String.concat(str: String): String =
asDynamic().concat(str)\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun String.match(regex: String):
Array<String>? = asDynamic().match(regex)\r\n\r\n//native public fun String.trim(): String\r\n//TODO:
String.replace to implement effective trimLeading and trimTrailing\r\n\r\n@kotlin.internal.InlineOnly\r\ninternal
inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern,
replacement)\r\n\r\n@SinceKotlin("1.2")\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUME
NTS")\r\npublic actual fun String.compareTo(other: String, ignoreCase: Boolean = false): Int {\r\n  if
(ignoreCase) {\r\n    val n1 = this.length\r\n    val n2 = other.length\r\n    val min = minOf(n1, n2)\r\n    if
(min == 0) return n1 - n2\r\n    var start = 0\r\n    while (true) {\r\n      val end = minOf(start + 16, min)\r\n
      var s1 = this.substring(start, end)\r\n      var s2 = other.substring(start, end)\r\n      if (s1 != s2) {\r\n
s1 = s1.toUpperCase()\r\n      s2 = s2.toUpperCase()\r\n      if (s1 != s2) {\r\n        s1 =
s1.toLowerCase()\r\n        s2 = s2.toLowerCase()\r\n        if (s1 != s2) {\r\n          return
s1.compareTo(s2)\r\n        }\r\n      }\r\n    }\r\n    if (end == min) break\r\n    start =
end\r\n  }\r\n  return n1 - n2\r\n} else {\r\n  return compareTo(other)\r\n}\r\n}\r\n\r\nprivate val
STRING_CASE_INSENSITIVE_ORDER = Comparator<String> { a, b -> a.compareTo(b, ignoreCase = true)

```



```

}

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER:
Comparator<String> get() = STRING_CASE_INSENSITIVE_ORDER
"/**
 * Copyright 2010-2019
JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
// Auto-generated file. DO NOT
EDIT!
package kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class
UByteArray
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@Published
Api
internal constructor(@PublishedApi internal val storage: ByteArray) : Collection<UByte> {
/**
 * Creates a new array of the specified [size], with all elements initialized to zero.
 */
public constructor(size: Int) :
this(ByteArray(size))
/**
 * Returns the array element at the given [index]. This method can be called
using the index operator.
 * If the [index] is out of bounds of this array, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public operator fun get(index: Int): UByte = storage[index].toUByte()
/**
 * Sets the element at the given
[index] to the given [value]. This method can be called using the index operator.
 * If the [index] is out
of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is
unspecified.
 */
public operator fun set(index: Int, value: UByte) {
storage[index] =
value.toByte()
}
/**
 * Returns the number of elements in the array.
 */
public override val size: Int
get() = storage.size
/**
 * Creates an iterator over the elements of the array.
 */
public override operator
fun iterator(): UByteIterator = Iterator(storage)
private class Iterator(private val array: ByteArray) :
UByteIterator() {
private var index = 0
override fun hasNext() = index < array.size
override
fun nextUByte() = if (index < array.size) array[index++].toUByte() else throw
NoSuchElementException(index.toString())
}
override fun contains(element: UByte): Boolean {
// TODO: Eliminate this check after KT-30016 gets fixed.
// Currently JS BE does not generate special
bridge method for this method.
if ((element as Any?) !is UByte) return false
return
storage.contains(element.toByte())
}
override fun containsAll(elements: Collection<UByte>):
Boolean {
return (elements as Collection<*>).all { it is UByte && storage.contains(it.toByte()) }
}
override fun isEmpty(): Boolean = this.storage.size == 0
}
/**
 * Creates a new array of the
specified [size], where each element is calculated by calling the specified
 * [init] function.
 * The
function [init] is called for each array element sequentially starting from the first one.
 * It should return the value
for an array element given its index.
 */
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray(size: Int, init: (Int) -> UByte): UByteArray {
return UByteArray(ByteArray(size) { index ->
init(index).toByte()
})
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public
inline fun ubyteArrayOf(vararg elements: UByte): UByteArray = elements
"/**
 * Copyright 2010-2019
JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
// Auto-generated file. DO NOT
EDIT!
package kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class
UIntArray
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@Published
Api
internal constructor(@PublishedApi internal val storage: IntArray) : Collection<UInt> {
/**
 * Creates
a new array of the specified [size], with all elements initialized to zero.
 */
public constructor(size: Int) :
this(IntArray(size))
/**
 * Returns the array element at the given [index]. This method can be called
using the index operator.
 * If the [index] is out of bounds of this array, throws an
[IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public operator fun get(index: Int): UInt = storage[index].toInt()
/**
 * Sets the element at the given [index]
to the given [value]. This method can be called using the index operator.
 * If the [index] is out of
bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is
unspecified.
 */
public operator fun set(index: Int, value: UInt) {
storage[index] = value.toInt()
}
/**
 * Returns the number of elements in the array.
 */
public override val size: Int
get() =

```



```

inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements\r\n", /*\r\n * Copyright 2010-2019
JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n// Auto-generated file. DO NOT
EDIT!\r\n\r\npackage kotlin\r\n\r\nimport
kotlin.experimental.\r\n\r\n@Suppress("\r\nNON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS\r\n")\r\n
\r\n@SinceKotlin("\r\n1.3\r\n")\r\n@ExperimentalUnsignedTypes\r\npublic inline class ULong @PublishedApi internal
constructor(@PublishedApi internal val data: Long) : Comparable<ULong> {\r\n\r\n    companion object {\r\n\r\n
/*\r\n\r\n        * A constant holding the minimum value an instance of ULong can have.\r\n\r\n        */\r\n\r\n        public
const val MIN_VALUE: ULong = ULong(0)\r\n\r\n\r\n        /*\r\n\r\n        * A constant holding the maximum value an
instance of ULong can have.\r\n\r\n        */\r\n\r\n        public const val MAX_VALUE: ULong = ULong(-1)\r\n\r\n\r\n
/*\r\n\r\n        * The number of bytes used to represent an instance of ULong in a binary form.\r\n\r\n        */\r\n\r\n
public const val SIZE_BYTES: Int = 8\r\n\r\n\r\n        /*\r\n\r\n        * The number of bits used to represent an instance of
ULong in a binary form.\r\n\r\n        */\r\n\r\n        public const val SIZE_BITS: Int = 64\r\n\r\n    }\r\n\r\n    /*\r\n\r\n
    * Compares this value with the specified value for order.\r\n\r\n    * Returns zero if this value is equal to the specified
other value, a negative number if it's less than other,\r\n\r\n    * or a positive number if it's greater than other.\r\n\r\n
    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun compareTo(other: UByte): Int =
this.compareTo(other.toULong())\r\n\r\n    /*\r\n\r\n    * Compares this value with the specified value for order.\r\n\r\n
    * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\r\n\r\n
    * or a positive number if it's greater than other.\r\n\r\n    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun
compareTo(other: UShort): Int = this.compareTo(other.toULong())\r\n\r\n    /*\r\n\r\n    * Compares this value with
the specified value for order.\r\n\r\n    * Returns zero if this value is equal to the specified other value, a negative
number if it's less than other,\r\n\r\n    * or a positive number if it's greater than other.\r\n\r\n    */\r\n\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun compareTo(other: UInt): Int =
this.compareTo(other.toULong())\r\n\r\n    /*\r\n\r\n    * Compares this value with the specified value for order.\r\n\r\n
    * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\r\n\r\n
    * or a positive number if it's greater than other.\r\n\r\n    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n
    @Suppress("\r\nOVERRIDE_BY_INLINE\r\n")\r\n    public override inline operator fun compareTo(other: ULong): Int
= ulongCompare(this.data, other.data)\r\n\r\n    /*\r\n\r\n    * Adds the other value to this value. *\r\n\r\n
    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun plus(other: UByte): ULong =
this.plus(other.toULong())\r\n\r\n    /*\r\n\r\n    * Adds the other value to this value. *\r\n\r\n    */\r\n\r\n
    @kotlin.internal.InlineOnly\r\n    public inline operator fun plus(other: UShort): ULong = this.plus(other.toULong())\r\n\r\n
    /*\r\n\r\n    * Adds the other value to this value. *\r\n\r\n    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n
    public inline operator fun plus(other: UInt): ULong = this.plus(other.toULong())\r\n\r\n    /*\r\n\r\n    * Adds the other value to this value. *\r\n\r\n
    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun plus(other: ULong): ULong = ULong(this.data.plus(other.data))\r\n\r\n
    /*\r\n\r\n    * Subtracts the other value from this value. *\r\n\r\n    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n
    public inline operator fun minus(other: UByte): ULong = this.minus(other.toULong())\r\n\r\n    /*\r\n\r\n    * Subtracts the other value from this value. *\r\n\r\n
    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun minus(other: UShort): ULong =
this.minus(other.toULong())\r\n\r\n    /*\r\n\r\n    * Subtracts the other value from this value. *\r\n\r\n
    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun minus(other: UInt): ULong =
this.minus(other.toULong())\r\n\r\n    /*\r\n\r\n    * Subtracts the other value from this value. *\r\n\r\n
    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun minus(other: ULong): ULong =
ULong(this.data.minus(other.data))\r\n\r\n    /*\r\n\r\n    * Multiplies this value by the other value. *\r\n\r\n
    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun times(other: UByte): ULong =
this.times(other.toULong())\r\n\r\n    /*\r\n\r\n    * Multiplies this value by the other value. *\r\n\r\n
    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun times(other: UShort): ULong =
this.times(other.toULong())\r\n\r\n    /*\r\n\r\n    * Multiplies this value by the other value. *\r\n\r\n
    */\r\n\r\n    @kotlin.internal.InlineOnly\r\n    public inline operator fun times(other: UInt): ULong =
this.times(other.toULong())\r\n\r\n    /*\r\n\r\n    * Multiplies this value by the other value. *\r\n\r\n
    */\r\n\r\n

```

```

@kotlin.internal.InlineOnly\r\n    public inline operator fun times(other: ULong): ULong =
    ULong(this.data.times(other.data))\r\n\r\n    /** Divides this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline operator fun div(other: UByte): ULong =
    this.div(other.toULong())\r\n    /** Divides this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline operator fun div(other: UShort): ULong = this.div(other.toULong())\r\n    /** Divides this value by the
    other value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline operator fun div(other: UInt): ULong =
    this.div(other.toULong())\r\n    /** Divides this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline operator fun div(other: ULong): ULong = ulongDivide(this, other)\r\n\r\n    /** Calculates the
    remainder of dividing this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline operator
    fun rem(other: UByte): ULong = this.rem(other.toULong())\r\n    /** Calculates the remainder of dividing this value
    by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline operator fun rem(other: UShort): ULong =
    this.rem(other.toULong())\r\n    /** Calculates the remainder of dividing this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline operator fun rem(other: UInt): ULong =
    this.rem(other.toULong())\r\n    /** Calculates the remainder of dividing this value by the other value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline operator fun rem(other: ULong): ULong = ulongRemainder(this,
    other)\r\n\r\n    /** Increments this value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline operator fun inc():
    ULong = ULong(data.inc())\r\n    /** Decrements this value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline
    operator fun dec(): ULong = ULong(data.dec())\r\n\r\n    /** Creates a range from this value to the specified [other]
    value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline operator fun rangeTo(other: ULong): ULongRange =
    ULongRange(this, other)\r\n\r\n    /** Shifts this value left by the [bitCount] number of bits. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline infix fun shl(bitCount: Int): ULong = ULong(data shl bitCount)\r\n
    /** Shifts this value right by the [bitCount] number of bits, filling the leftmost bits with zeros. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline infix fun shr(bitCount: Int): ULong = ULong(data ushr bitCount)\r\n
    /** Performs a bitwise AND operation between the two values. */\r\n
@kotlin.internal.InlineOnly\r\n    public
    inline infix fun and(other: ULong): ULong = ULong(this.data and other.data)\r\n    /** Performs a bitwise OR
    operation between the two values. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline infix fun or(other: ULong):
    ULong = ULong(this.data or other.data)\r\n    /** Performs a bitwise XOR operation between the two values. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline infix fun xor(other: ULong): ULong = ULong(this.data xor
    other.data)\r\n    /** Inverts the bits in this value. */\r\n
@kotlin.internal.InlineOnly\r\n    public inline fun inv():
    ULong = ULong(data.inv())\r\n\r\n    /** * Converts this [ULong] value to [Byte].\r\n    * If this value
    is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents\r\n    * the same numerical
    value as this `ULong`.\r\n    * The resulting `Byte` value is represented by the least significant 8 bits of this
    `ULong` value.\r\n    * Note that the resulting `Byte` value may be negative.\r\n    */\r\n
@kotlin.internal.InlineOnly\r\n    public inline fun toByte(): Byte = data.toByte()\r\n    /**\r\n    * Converts this
    [ULong] value to [Short].\r\n    * If this value is less than or equals to [Short.MAX_VALUE], the resulting
    `Short` value represents\r\n    * the same numerical value as this `ULong`.\r\n    * The resulting `Short`
    value is represented by the least significant 16 bits of this `ULong` value.\r\n    * Note that the resulting `Short`
    value may be negative.\r\n    */\r\n
@kotlin.internal.InlineOnly\r\n    public inline fun toShort(): Short =
    data.toShort()\r\n    /**\r\n    * Converts this [ULong] value to [Int].\r\n    * If this value is less than or
    equals to [Int.MAX_VALUE], the resulting `Int` value represents\r\n    * the same numerical value as this
    `ULong`.\r\n    * The resulting `Int` value is represented by the least significant 32 bits of this `ULong`
    value.\r\n    * Note that the resulting `Int` value may be negative.\r\n    */\r\n
@kotlin.internal.InlineOnly\r\n    public inline fun toInt(): Int = data.toInt()\r\n    /**\r\n    * Converts this [ULong] value to [Long].\r\n    * If
    this value is less than or equals to [Long.MAX_VALUE], the resulting `Long` value represents\r\n    * the same
    numerical value as this `ULong`. Otherwise the result is negative.\r\n    * The resulting `Long` value has the
    same binary representation as this `ULong` value.\r\n    */\r\n
@kotlin.internal.InlineOnly\r\n    public inline fun
    toLong(): Long = data\r\n\r\n    /**\r\n    * Converts this [ULong] value to [UByte].\r\n    * If this value is
    less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents\r\n    * the same numerical
  
```

```

value as this `ULong`.r\n *r\n * The resulting `UByte` value is represented by the least significant 8 bits of
this `ULong` value.r\n */r\n @kotlin.internal.InlineOnlyr\n public inline fun toUByte(): UByte =
data.toUByte()r\n /**r\n * Converts this [ULong] value to [UShort].r\n *r\n * If this value is less than or
equals to [UShort.MAX_VALUE], the resulting `UShort` value representsr\n * the same numerical value as this
`ULong`.r\n *r\n * The resulting `UShort` value is represented by the least significant 16 bits of this `ULong`
value.r\n */r\n @kotlin.internal.InlineOnlyr\n public inline fun toUShort(): UShort = data.toUShort()r\n
/**r\n * Converts this [ULong] value to [UInt].r\n *r\n * If this value is less than or equals to
[UInt.MAX_VALUE], the resulting `UInt` value representsr\n * the same numerical value as this `ULong`.r\n
*r\n * The resulting `UInt` value is represented by the least significant 32 bits of this `ULong` value.r\n */r\n
@kotlin.internal.InlineOnlyr\n public inline fun toUInt(): UInt = data.toUInt()r\n /** Returns this value. */r\n
@kotlin.internal.InlineOnlyr\n public inline fun toULong(): ULong = thisr\nr\n /**r\n * Converts this
[ULong] value to [Float].r\n *r\n * The resulting value is the closest `Float` to this `ULong` value.r\n * In
case when this `ULong` value is exactly between two `Float`s,r\n * the one with zero at least significant bit of
mantissa is selected.r\n */r\n @kotlin.internal.InlineOnlyr\n public inline fun toFloat(): Float =
this.toDouble().toFloat()r\n /**r\n * Converts this [ULong] value to [Double].r\n *r\n * The resulting
value is the closest `Double` to this `ULong` value.r\n * In case when this `ULong` value is exactly between two
`Double`s,r\n * the one with zero at least significant bit of mantissa is selected.r\n */r\n
@kotlin.internal.InlineOnlyr\n public inline fun toDouble(): Double = ulongToDouble(data)r\nr\n public
override fun toString(): String = ulongToString(data)r\nr\nr\n /**r\n * Converts this [Byte] value to
[ULong].r\n *r\n * If this value is positive, the resulting `ULong` value represents the same numerical value as this
`Byte`.r\n *r\n * The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `Byte`
value,r\n * whereas the most significant 56 bits are filled with the sign bit of this value.r\n
*/r\n@SinceKotlin("1.3")r\n@ExperimentalUnsignedTypesr\n@kotlin.internal.InlineOnlyr\npublic inline fun
Byte.toULong(): ULong = ULong(this.toLong())r\n/**r\n * Converts this [Short] value to [ULong].r\n *r\n * If
this value is positive, the resulting `ULong` value represents the same numerical value as this `Short`.r\n *r\n * The
least significant 16 bits of the resulting `ULong` value are the same as the bits of this `Short` value,r\n * whereas
the most significant 48 bits are filled with the sign bit of this value.r\n
*/r\n@SinceKotlin("1.3")r\n@ExperimentalUnsignedTypesr\n@kotlin.internal.InlineOnlyr\npublic inline fun
Short.toULong(): ULong = ULong(this.toLong())r\n/**r\n * Converts this [Int] value to [ULong].r\n *r\n * If this
value is positive, the resulting `ULong` value represents the same numerical value as this `Int`.r\n *r\n * The least
significant 32 bits of the resulting `ULong` value are the same as the bits of this `Int` value,r\n * whereas the most
significant 32 bits are filled with the sign bit of this value.r\n
*/r\n@SinceKotlin("1.3")r\n@ExperimentalUnsignedTypesr\n@kotlin.internal.InlineOnlyr\npublic inline fun
Int.toULong(): ULong = ULong(this.toLong())r\n/**r\n * Converts this [Long] value to [ULong].r\n *r\n * If this
value is positive, the resulting `ULong` value represents the same numerical value as this `Long`.r\n *r\n * The
resulting `ULong` value has the same binary representation as this `Long` value.r\n
*/r\n@SinceKotlin("1.3")r\n@ExperimentalUnsignedTypesr\n@kotlin.internal.InlineOnlyr\npublic inline fun
Long.toULong(): ULong = ULong(this)r\nr\n/**r\n * Converts this [Float] value to [ULong].r\n *r\n * The
fractional part, if any, is rounded down.r\n * Returns zero if this `Float` value is negative or `NaN`,
[ULong.MAX_VALUE] if it's bigger than `ULong.MAX_VALUE`.r\n
*/r\n@SinceKotlin("1.3")r\n@ExperimentalUnsignedTypesr\n@kotlin.internal.InlineOnlyr\npublic inline fun
Float.toULong(): ULong = doubleToULong(this.toDouble())r\n/**r\n * Converts this [Double] value to
[ULong].r\n *r\n * The fractional part, if any, is rounded down.r\n * Returns zero if this `Double` value is negative
or `NaN`, [ULong.MAX_VALUE] if it's bigger than `ULong.MAX_VALUE`.r\n
*/r\n@SinceKotlin("1.3")r\n@ExperimentalUnsignedTypesr\n@kotlin.internal.InlineOnlyr\npublic inline fun
Double.toULong(): ULong = doubleToULong(this)r\n","/**r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin
Programming Language contributors.r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.r\n */r\nr\n// Auto-generated file. DO NOT EDIT!r\nr\npackage

```


ULongArray.component2(): ULong { \n return get(1)\n}\n\n/**\n * Returns 2nd *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UByteArray.component2(): UByte { \n return get(1)\n}\n\n/**\n * Returns 2nd *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UShortArray.component2(): UShort { \n return get(1)\n}\n\n/**\n * Returns 3rd *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UIntArray.component3(): UInt { \n return get(2)\n}\n\n/**\n * Returns 3rd *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

ULongArray.component3(): ULong { \n return get(2)\n}\n\n/**\n * Returns 3rd *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UByteArray.component3(): UByte { \n return get(2)\n}\n\n/**\n * Returns 3rd *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UShortArray.component3(): UShort { \n return get(2)\n}\n\n/**\n * Returns 4th *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UIntArray.component4(): UInt { \n return get(3)\n}\n\n/**\n * Returns 4th *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

ULongArray.component4(): ULong { \n return get(3)\n}\n\n/**\n * Returns 4th *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UByteArray.component4(): UByte { \n return get(3)\n}\n\n/**\n * Returns 4th *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UShortArray.component4(): UShort { \n return get(3)\n}\n\n/**\n * Returns 5th *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UIntArray.component5(): UInt { \n return get(4)\n}\n\n/**\n * Returns 5th *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

ULongArray.component5(): ULong { \n return get(4)\n}\n\n/**\n * Returns 5th *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UByteArray.component5(): UByte { \n return get(4)\n}\n\n/**\n * Returns 5th *element* from the collection.\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun

UShortArray.component5(): UShort { \n return get(4)\n}\n\n/**\n * Returns an element at the given [index] or
 throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n * \n * @sample
 samples.collections.Collections.Elements.elementAt\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun UIntArray.elementAt(index: Int):
 UInt\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is
 out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun ULongArray.elementAt(index: Int):
 ULong\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index]
 is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun UByteArray.elementAt(index: Int):
 UByte\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index]
 is out of bounds of this array.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun UShortArray.elementAt(index: Int):
 UShort\n\n/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
 [index] is out of bounds of this array.\n * \n * @sample
 samples.collections.Collections.Elements.elementAtOrElse\n
 \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
 UIntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UInt): UInt { \n return if (index >= 0 && index <=
 lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
 calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample

```

samples.collections.Collections.Elements.elementAtOrElse\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {\n    return if (index >= 0 &&
index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the
result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {\n    return if (index >= 0 && index
<= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result
of calling the [defaultValue] function if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrElse\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {\n    return if (index >= 0 &&
index <= lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or
`null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.elementAtOrNull(index: Int): UInt? {\n    return this.getOrNull(index)\n}\n\n/**\n * Returns an element
at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.elementAtOrNull(index: Int): ULong? {\n    return this.getOrNull(index)\n}\n\n/**\n * Returns an
element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.elementAtOrNull(index: Int): UByte? {\n    return this.getOrNull(index)\n}\n\n/**\n * Returns an
element at the given [index] or `null` if the [index] is out of bounds of this array.\n * \n * @sample
samples.collections.Collections.Elements.elementAtOrNull\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.elementAtOrNull(index: Int): UShort? {\n    return this.getOrNull(index)\n}\n\n/**\n * Returns the
first element matching the given [predicate], or `null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.find(predicate: (UInt) -> Boolean): UInt? {\n    return firstOrNull(predicate)\n}\n\n/**\n * Returns the
first element matching the given [predicate], or `null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.find(predicate: (ULong) -> Boolean): ULong? {\n    return firstOrNull(predicate)\n}\n\n/**\n *
Returns the first element matching the given [predicate], or `null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.find(predicate: (UByte) -> Boolean): UByte? {\n    return firstOrNull(predicate)\n}\n\n/**\n * Returns
the first element matching the given [predicate], or `null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.find(predicate: (UShort) -> Boolean): UShort? {\n    return firstOrNull(predicate)\n}\n\n/**\n *
Returns the last element matching the given [predicate], or `null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.findLast(predicate: (UInt) -> Boolean): UInt? {\n    return lastOrNull(predicate)\n}\n\n/**\n * Returns the
last element matching the given [predicate], or `null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.findLast(predicate: (ULong) -> Boolean): ULong? {\n    return lastOrNull(predicate)\n}\n\n/**\n *

```


Returns the last element matching the given [predicate], or `null` if no such element was found.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.findLast(predicate: (UByte) -> Boolean): UByte? {\n    return lastOrNull(predicate)\n}\n\n/**\n *
Returns the last element matching the given [predicate], or `null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.findLast(predicate: (UShort) -> Boolean): UShort? {\n    return lastOrNull(predicate)\n}\n\n/**\n *
Returns first element.\n * @throws [NoSuchElementException] if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.first(): UInt {\n    return storage.first().toUInt()\n}\n\n/**\n * Returns first element.\n * @throws
[NoSuchElementException] if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.first(): ULong {\n    return storage.first().toULong()\n}\n\n/**\n * Returns first element.\n * @throws
[NoSuchElementException] if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.first(): UByte {\n    return storage.first().toUByte()\n}\n\n/**\n * Returns first element.\n * @throws
[NoSuchElementException] if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.first(): UShort {\n    return storage.first().toUShort()\n}\n\n/**\n * Returns the first element matching
the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.first(predicate: (UInt) -> Boolean): UInt {\n    for (element in this) if (predicate(element)) return
element\n    throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n *
Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such
element is found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.first(predicate: (ULong) -> Boolean): ULong {\n    for (element in this) if (predicate(element)) return
element\n    throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n *
Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such
element is found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.first(predicate: (UByte) -> Boolean): UByte {\n    for (element in this) if (predicate(element)) return
element\n    throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n *
Returns the first element matching the given [predicate].\n * @throws [NoSuchElementException] if no such
element is found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.first(predicate: (UShort) -> Boolean): UShort {\n    for (element in this) if (predicate(element)) return
element\n    throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n *
Returns the first element, or `null` if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.firstOrNull(): UInt? {\n    return
if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.firstOrNull(): ULong? {\n    return
if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.firstOrNull(): UByte? {\n    return
if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element, or `null` if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.firstOrNull(): UShort? {\n    return
if (isEmpty()) null else this[0]\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null`
if element was not found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

```

UIntArray.firstOrNull(predicate: (UInt) -> Boolean): UInt? {\n  for (element in this) if (predicate(element)) return
element\n  return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if element
was not found.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic
inline fun ULongArray.firstOrNull(predicate: (ULong) -> Boolean): ULong? {\n  for (element in this) if
(predicate(element)) return element\n  return null\n}\n\n/**\n * Returns the first element matching the given
[predicate], or `null` if element was not found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.firstOrNull(predicate: (UByte) -> Boolean): UByte? {\n  for (element in this) if (predicate(element))
return element\n  return null\n}\n\n/**\n * Returns the first element matching the given [predicate], or `null` if
element was not found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.firstOrNull(predicate: (UShort) -> Boolean): UShort? {\n  for (element in this) if (predicate(element))
return element\n  return null\n}\n\n/**\n * Returns an element at the given [index] or the result of calling the
[defaultValue] function if the [index] is out of bounds of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.getOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.getOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.getOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or the result of
calling the [defaultValue] function if the [index] is out of bounds of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.getOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {\n  return if (index >= 0 && index <=
lastIndex) get(index) else defaultValue(index)\n}\n\n/**\n * Returns an element at the given [index] or `null` if the
[index] is out of bounds of this array.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
UIntArray.getOrNull(index: Int): UInt? {\n  return if (index >= 0 && index <= lastIndex) get(index) else
null\n}\n\n/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.getOrNull(index: Int):
ULong? {\n  return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at
the given [index] or `null` if the [index] is out of bounds of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.getOrNull(index: Int): UByte?
{\n  return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns an element at the
given [index] or `null` if the [index] is out of bounds of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.getOrNull(index: Int):
UShort? {\n  return if (index >= 0 && index <= lastIndex) get(index) else null\n}\n\n/**\n * Returns first index of
[element], or -1 if the array does not contain element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.indexOf(element: UInt): Int {\n  return storage.indexOf(element.toInt())\n}\n\n/**\n * Returns first
index of [element], or -1 if the array does not contain element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.indexOf(element: ULong): Int {\n  return storage.indexOf(element.toLong())\n}\n\n/**\n * Returns
first index of [element], or -1 if the array does not contain element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

```

```

UByteArray.indexOf(element: UByte): Int {\n  return storage.indexOf(element.toByte())\n}\n\n/**\n * Returns first index of [element], or -1 if the array does not contain element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.indexOf(element: UShort): Int {\n  return storage.indexOf(element.toShort())\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.indexOfFirst(predicate: (UInt) -> Boolean): Int {\n  return storage.indexOfFirst { predicate(it.toUInt()) }\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.indexOfFirst(predicate: (ULong) -> Boolean): Int {\n  return storage.indexOfFirst { predicate(it.toULong()) }\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int {\n  return storage.indexOfFirst { predicate(it.toUByte()) }\n}\n\n/**\n * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.indexOfFirst(predicate: (UShort) -> Boolean): Int {\n  return storage.indexOfFirst { predicate(it.toUShort()) }\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.indexOfLast(predicate: (UInt) -> Boolean): Int {\n  return storage.indexOfLast { predicate(it.toUInt()) }\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.indexOfLast(predicate: (ULong) -> Boolean): Int {\n  return storage.indexOfLast { predicate(it.toULong()) }\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.indexOfLast(predicate: (UByte) -> Boolean): Int {\n  return storage.indexOfLast { predicate(it.toUByte()) }\n}\n\n/**\n * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.indexOfLast(predicate: (UShort) -> Boolean): Int {\n  return storage.indexOfLast { predicate(it.toUShort()) }\n}\n\n/**\n * Returns the last element.\n * @throws [NoSuchElementException] if the array is empty.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.last(): UInt {\n  return storage.last().toUInt()\n}\n\n/**\n * Returns the last element.\n * @throws [NoSuchElementException] if the array is empty.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.last(): ULong {\n  return storage.last().toULong()\n}\n\n/**\n * Returns the last element.\n * @throws [NoSuchElementException] if the array is empty.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.last(): UByte {\n  return storage.last().toUByte()\n}\n\n/**\n * Returns the last element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.last(predicate: (UInt) -> Boolean): UInt {\n  for (index in this.indices.reversed()) {\n    val element =

```

```

this[index]\n    if (predicate(element)) return element\n    }\n    throw NoSuchElementException("Array contains
no element matching the predicate.")\n}\n\n/**\n * Returns the last element matching the given [predicate].\n *
@throws [NoSuchElementException] if no such element is found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.last(predicate: (ULong) -> Boolean): ULong {\n    for (index in this.indices.reversed()) {\n        val
element = this[index]\n        if (predicate(element)) return element\n    }\n    throw
NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns the last
element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.last(predicate: (UByte) -> Boolean): UByte {\n    for (index in this.indices.reversed()) {\n        val
element = this[index]\n        if (predicate(element)) return element\n    }\n    throw
NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns the last
element matching the given [predicate].\n * @throws [NoSuchElementException] if no such element is found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.last(predicate: (UShort) -> Boolean): UShort {\n    for (index in this.indices.reversed()) {\n        val
element = this[index]\n        if (predicate(element)) return element\n    }\n    throw
NoSuchElementException("Array contains no element matching the predicate.")\n}\n\n/**\n * Returns last index
of [element], or -1 if the array does not contain element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.lastIndexOf(element: UInt): Int {\n    return storage.lastIndexOf(element.toInt())\n}\n\n/**\n * Returns
last index of [element], or -1 if the array does not contain element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.lastIndexOf(element: ULong): Int {\n    return storage.lastIndexOf(element.toLong())\n}\n\n/**\n *
Returns last index of [element], or -1 if the array does not contain element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.lastIndexOf(element: UByte): Int {\n    return storage.lastIndexOf(element.toByte())\n}\n\n/**\n *
Returns last index of [element], or -1 if the array does not contain element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.lastIndexOf(element: UShort): Int {\n    return storage.lastIndexOf(element.toShort())\n}\n\n/**\n *
Returns the last element, or `null` if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.lastOrNull(): UInt? {\n    return
if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.lastOrNull(): ULong? {\n    return
if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.lastOrNull(): UByte? {\n    return
if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element, or `null` if the array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.lastOrNull(): UShort? {\n    return
if (isEmpty()) null else this[size - 1]\n}\n\n/**\n * Returns the last element matching the given [predicate], or
`null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.lastOrNull(predicate: (UInt) -> Boolean): UInt? {\n    for (index in this.indices.reversed()) {\n        val
element = this[index]\n        if (predicate(element)) return element\n    }\n    return null\n}\n\n/**\n * Returns the last
element matching the given [predicate], or `null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.lastOrNull(predicate: (ULong) -> Boolean): ULong? {\n    for (index in this.indices.reversed()) {\n        val
element = this[index]\n        if (predicate(element)) return element\n    }\n    return null\n}\n\n/**\n * Returns the
last element matching the given [predicate], or `null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

```

```

ByteArray.lastOrNull(predicate: (UByte) -> Boolean): UByte? {\n  for (index in this.indices.reversed()) {\n
val element = this[index]\n    if (predicate(element)) return element\n  }\n  return null\n}\n\n/**\n * Returns the
last element matching the given [predicate], or `null` if no such element was found.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.lastOrNull(predicate: (UShort) -> Boolean): UShort? {\n  for (index in this.indices.reversed()) {\n
val element = this[index]\n    if (predicate(element)) return element\n  }\n  return null\n}\n\n/**\n * Returns a
random element from this array.\n * \n * @throws NoSuchElementException if this array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.random(): UInt {\n  return random(Random)\n}\n\n/**\n * Returns a random element from this array.\n
* \n * @throws NoSuchElementException if this array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.random(): ULong {\n  return random(Random)\n}\n\n/**\n * Returns a random element from this
array.\n * \n * @throws NoSuchElementException if this array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.random(): UByte {\n  return random(Random)\n}\n\n/**\n * Returns a random element from this
array.\n * \n * @throws NoSuchElementException if this array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.random(): UShort {\n  return random(Random)\n}\n\n/**\n * Returns a random element from this
array using the specified source of randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.random(random: Random): UInt
{\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return
get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of
randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.random(random: Random):
ULong {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return
get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of
randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.random(random: Random):
UByte {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return
get(random.nextInt(size))\n}\n\n/**\n * Returns a random element from this array using the specified source of
randomness.\n * \n * @throws NoSuchElementException if this array is empty.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.random(random: Random):
UShort {\n  if (isEmpty())\n    throw NoSuchElementException("Array is empty.")\n  return
get(random.nextInt(size))\n}\n\n/**\n * Returns the single element, or throws an exception if the array is empty or
has more than one element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.single(): UInt {\n  return storage.single().toUInt()\n}\n\n/**\n * Returns the single element, or throws an
exception if the array is empty or has more than one element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.single(): ULong {\n  return storage.single().toULong()\n}\n\n/**\n * Returns the single element, or
throws an exception if the array is empty or has more than one element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.single(): UByte {\n  return storage.single().toUByte()\n}\n\n/**\n * Returns the single element, or
throws an exception if the array is empty or has more than one element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.single(): UShort {\n  return storage.single().toUShort()\n}\n\n/**\n * Returns the single element
matching the given [predicate], or throws exception if there is no or more than one matching element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

```

```

UIntArray.single(predicate: (UInt) -> Boolean): UInt {
    var single: UInt? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
}
@Suppress("UNCHECKED_CAST")
return single as UInt
}

* Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.

* \n @SinceKotlin("1.3") \n @ExperimentalUnsignedTypes \n @kotlin.internal.InlineOnly \n public inline fun
ULongArray.single(predicate: (ULong) -> Boolean): ULong {
    var single: ULong? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
}
@Suppress("UNCHECKED_CAST")
return single as ULong
}

* Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.

* \n @SinceKotlin("1.3") \n @ExperimentalUnsignedTypes \n @kotlin.internal.InlineOnly \n public inline fun
UByteArray.single(predicate: (UByte) -> Boolean): UByte {
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
}
@Suppress("UNCHECKED_CAST")
return single as UByte
}

* Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.

* \n @SinceKotlin("1.3") \n @ExperimentalUnsignedTypes \n @kotlin.internal.InlineOnly \n public inline fun
UShortArray.single(predicate: (UShort) -> Boolean): UShort {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
}
@Suppress("UNCHECKED_CAST")
return single as UShort
}

* Returns single element, or `null` if the array is empty or has more than one element.

* \n @SinceKotlin("1.3") \n @ExperimentalUnsignedTypes \n public fun UIntArray.singleOrNull(): UInt? {
    return if (size == 1) this[0] else null
}

* Returns single element, or `null` if the array is empty or has more than one element.

* \n @SinceKotlin("1.3") \n @ExperimentalUnsignedTypes \n public fun
ULongArray.singleOrNull(): ULong? {
    return if (size == 1) this[0] else null
}

* Returns single element, or `null` if the array is empty or has more than one element.

* \n @SinceKotlin("1.3") \n @ExperimentalUnsignedTypes \n public fun
UByteArray.singleOrNull(): UByte? {
    return if (size == 1) this[0] else null
}

* Returns single element, or `null` if the array is empty or has more than one element.

* \n @SinceKotlin("1.3") \n @ExperimentalUnsignedTypes \n public fun
UShortArray.singleOrNull(): UShort? {
    return if (size == 1) this[0] else null
}

* Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

* \n @SinceKotlin("1.3") \n @ExperimentalUnsignedTypes \n @kotlin.internal.InlineOnly \n public inline fun
UIntArray.singleOrNull(predicate: (UInt) -> Boolean): UInt? {
    var single: UInt? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

* Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

* \n @SinceKotlin("1.3") \n @ExperimentalUnsignedTypes \n @kotlin.internal.InlineOnly \n public inline fun
ULongArray.singleOrNull(predicate: (ULong) -> Boolean): ULong? {
    var single: ULong? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

* Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

* \n @SinceKotlin("1.3") \n @ExperimentalUnsignedTypes \n @kotlin.internal.InlineOnly \n public inline

```

```

fun UByteArray.singleOrNull(predicate: (UByte) -> Boolean): UByte? {
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/** Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 *
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public inline fun UShortArray.singleOrNull(predicate: (UShort) -> Boolean): UShort? {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/** Returns a list containing all elements except first [n] elements.
 *
 * @throws IllegalArgumentException if [n] is negative.
 *
 * @sample samples.collections.Collections.Transformations.drop
 *
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * public fun UIntArray.drop(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/** Returns a list containing all elements except first [n] elements.
 *
 * @throws IllegalArgumentException if [n] is negative.
 *
 * @sample samples.collections.Collections.Transformations.drop
 *
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * public fun ULongArray.drop(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/** Returns a list containing all elements except first [n] elements.
 *
 * @throws IllegalArgumentException if [n] is negative.
 *
 * @sample samples.collections.Collections.Transformations.drop
 *
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * public fun UByteArray.drop(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/** Returns a list containing all elements except first [n] elements.
 *
 * @throws IllegalArgumentException if [n] is negative.
 *
 * @sample samples.collections.Collections.Transformations.drop
 *
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * public fun UShortArray.drop(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/** Returns a list containing all elements except last [n] elements.
 *
 * @throws IllegalArgumentException if [n] is negative.
 *
 * @sample samples.collections.Collections.Transformations.drop
 *
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * public fun UIntArray.dropLast(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/** Returns a list containing all elements except last [n] elements.
 *
 * @throws IllegalArgumentException if [n] is negative.
 *
 * @sample samples.collections.Collections.Transformations.drop
 *
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * public fun ULongArray.dropLast(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/** Returns a list containing all elements except last [n] elements.
 *
 * @throws IllegalArgumentException if [n] is negative.
 *
 * @sample samples.collections.Collections.Transformations.drop
 *
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * public fun UByteArray.dropLast(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/** Returns a list containing all elements except last [n] elements.
 *
 * @throws IllegalArgumentException if [n] is negative.
 *
 * @sample samples.collections.Collections.Transformations.drop
 *
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * public fun UShortArray.dropLast(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/** Returns a list containing all elements except last elements that satisfy the given

```

```

[predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.dropLastWhile(predicate: (UInt) -> Boolean): List<UInt> {\n  for (index in lastIndex downTo 0) {\n
if (!predicate(this[index])) {\n      return take(index + 1)\n  } \n } \n return emptyList()\n}\n\n/**\n *
Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.dropLastWhile(predicate: (ULong) -> Boolean): List<ULong> {\n  for (index in lastIndex downTo 0)
{\n  if (!predicate(this[index])) {\n      return take(index + 1)\n  } \n } \n return emptyList()\n}\n\n/**\n *
Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.dropLastWhile(predicate: (UByte) -> Boolean): List<UByte> {\n  for (index in lastIndex downTo 0)
{\n  if (!predicate(this[index])) {\n      return take(index + 1)\n  } \n } \n return emptyList()\n}\n\n/**\n *
Returns a list containing all elements except last elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.dropLastWhile(predicate: (UShort) -> Boolean): List<UShort> {\n  for (index in lastIndex downTo
0) {\n  if (!predicate(this[index])) {\n      return take(index + 1)\n  } \n } \n return
emptyList()\n}\n\n/**\n * Returns a list containing all elements except first elements that satisfy the given
[predicate].\n * \n * @sample samples.collections.Collections.Transformations.drop\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.dropWhile(predicate: (UInt) -> Boolean): List<UInt> {\n  var yielding = false\n  val list =
ArrayList<UInt>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item)) {\n
      list.add(item)\n      yielding = true\n    } \n return list\n}\n\n/**\n * Returns a list containing all
elements except first elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.dropWhile(predicate: (ULong) -> Boolean): List<ULong> {\n  var yielding = false\n  val list =
ArrayList<ULong>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item))
{\n      list.add(item)\n      yielding = true\n    } \n return list\n}\n\n/**\n * Returns a list containing all
elements except first elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.dropWhile(predicate: (UByte) -> Boolean): List<UByte> {\n  var yielding = false\n  val list =
ArrayList<UByte>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item))
{\n      list.add(item)\n      yielding = true\n    } \n return list\n}\n\n/**\n * Returns a list containing all
elements except first elements that satisfy the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.drop\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.dropWhile(predicate: (UShort) -> Boolean): List<UShort> {\n  var yielding = false\n  val list =
ArrayList<UShort>()\n  for (item in this)\n    if (yielding)\n      list.add(item)\n    else if (!predicate(item))
{\n      list.add(item)\n      yielding = true\n    } \n return list\n}\n\n/**\n * Returns a list containing only
elements matching the given [predicate].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.filter(predicate: (UInt) -> Boolean): List<UInt> {\n  return filterTo(ArrayList<UInt>(),
predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

```



```

ULongArray.filter(predicate: (ULong) -> Boolean): List<ULong> {\n  return filterTo(ArrayList<ULong>(),
predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.filter(predicate: (UByte) -> Boolean): List<UByte> {\n  return filterTo(ArrayList<UByte>(),
predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {\n  return filterTo(ArrayList<UShort>(),
predicate)\n}\n\n/**\n * Returns a list containing only elements matching the given [predicate].\n * @param
[predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate
evaluation on the element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.filterIndexed(predicate: (index: Int, UInt) -> Boolean): List<UInt> {\n  return
filterIndexedTo(ArrayList<UInt>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the
given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and
returns the result of predicate evaluation on the element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.filterIndexed(predicate: (index: Int, ULong) -> Boolean): List<ULong> {\n  return
filterIndexedTo(ArrayList<ULong>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the
given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and
returns the result of predicate evaluation on the element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.filterIndexed(predicate: (index: Int, UByte) -> Boolean): List<UByte> {\n  return
filterIndexedTo(ArrayList<UByte>(), predicate)\n}\n\n/**\n * Returns a list containing only elements matching the
given [predicate].\n * @param [predicate] function that takes the index of an element and the element itself\n * and
returns the result of predicate evaluation on the element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.filterIndexed(predicate: (index: Int, UShort) -> Boolean): List<UShort> {\n  return
filterIndexedTo(ArrayList<UShort>(), predicate)\n}\n\n/**\n * Appends all elements matching the given [predicate]
to the given [destination].\n * @param [predicate] function that takes the index of an element and the element
itself\n * and returns the result of predicate evaluation on the element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in UInt>> UIntArray.filterIndexedTo(destination: C, predicate: (index: Int, UInt) -> Boolean): C
{\n  forEachIndexed { index, element ->\n    if (predicate(index, element)) destination.add(element)\n  }\n  return
destination}\n}\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n *
@param [predicate] function that takes the index of an element and the element itself\n * and returns the result of
predicate evaluation on the element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in ULong>> ULongArray.filterIndexedTo(destination: C, predicate: (index: Int, ULong) ->
Boolean): C {\n  forEachIndexed { index, element ->\n    if (predicate(index, element))
destination.add(element)\n  }\n  return destination}\n}\n\n/**\n * Appends all elements matching the given
[predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the
element itself\n * and returns the result of predicate evaluation on the element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in UByte>> UByteArray.filterIndexedTo(destination: C, predicate: (index: Int, UByte) ->
Boolean): C {\n  forEachIndexed { index, element ->\n    if (predicate(index, element))
destination.add(element)\n  }\n  return destination}\n}\n\n/**\n * Appends all elements matching the given
[predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the
element itself\n * and returns the result of predicate evaluation on the element.\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in UShort>> UShortArray.filterIndexedTo(destination: C, predicate: (index: Int, UShort) ->
Boolean): C {\n  forEachIndexed { index, element ->\n    if (predicate(index, element))
destination.add(element)\n  }\n  return destination\n}\n\n/**\n * Returns a list containing all elements not
matching the given [predicate].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.filterNot(predicate: (UInt) -> Boolean): List<UInt> {\n  return filterNotTo(ArrayList<UInt>(),
predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.filterNot(predicate: (ULong) -> Boolean): List<ULong> {\n  return filterNotTo(ArrayList<ULong>(),
predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.filterNot(predicate: (UByte) -> Boolean): List<UByte> {\n  return filterNotTo(ArrayList<UByte>(),
predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.filterNot(predicate: (UShort) -> Boolean): List<UShort> {\n  return
filterNotTo(ArrayList<UShort>(), predicate)\n}\n\n/**\n * Appends all elements not matching the given [predicate]
to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in UInt>> UIntArray.filterNotTo(destination: C, predicate: (UInt) -> Boolean): C {\n  for
(element in this) if (!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements not matching the given [predicate] to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in ULong>> ULongArray.filterNotTo(destination: C, predicate: (ULong) -> Boolean): C {\n  for
(element in this) if (!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends
all elements not matching the given [predicate] to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in UByte>> UByteArray.filterNotTo(destination: C, predicate: (UByte) -> Boolean): C {\n  for
(element in this) if (!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements not matching the given [predicate] to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in UShort>> UShortArray.filterNotTo(destination: C, predicate: (UShort) -> Boolean): C {\n  for
(element in this) if (!predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends
all elements matching the given [predicate] to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in UInt>> UIntArray.filterTo(destination: C, predicate: (UInt) -> Boolean): C {\n  for (element
in this) if (predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all elements
matching the given [predicate] to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in ULong>> ULongArray.filterTo(destination: C, predicate: (ULong) -> Boolean): C {\n  for
(element in this) if (predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements matching the given [predicate] to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in UByte>> UByteArray.filterTo(destination: C, predicate: (UByte) -> Boolean): C {\n  for
(element in this) if (predicate(element)) destination.add(element)\n  return destination\n}\n\n/**\n * Appends all
elements matching the given [predicate] to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C :
MutableCollection<in UShort>> UShortArray.filterTo(destination: C, predicate: (UShort) -> Boolean): C {\n  for

```

(element in this) if (predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.slice(indices: IntRange):
List<UInt> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive +
1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.slice(indices: IntRange):
List<ULong> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive
+ 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.slice(indices: IntRange):
List<UByte> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive
+ 1).asList()\n}\n\n/**\n * Returns a list containing elements at indices in the specified [indices] range.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.slice(indices: IntRange):
List<UShort> {\n if (indices.isEmpty()) return listOf()\n return copyOfRange(indices.start, indices.endInclusive
+ 1).asList()\n}\n\n/**\n * Returns a list containing elements at specified [indices].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.slice(indices: Iterable<Int>):
List<UInt> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list =
ArrayList<UInt>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n *
Returns a list containing elements at specified [indices].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.slice(indices: Iterable<Int>):
List<ULong> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list =
ArrayList<ULong>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n *
Returns a list containing elements at specified [indices].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.slice(indices: Iterable<Int>):
List<UByte> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list =
ArrayList<UByte>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n *
Returns a list containing elements at specified [indices].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.slice(indices: Iterable<Int>):
List<UShort> {\n val size = indices.collectionSizeOrDefault(10)\n if (size == 0) return emptyList()\n val list =
ArrayList<UShort>(size)\n for (index in indices) {\n list.add(get(index))\n }\n return list\n}\n\n/**\n *
Returns an array containing elements of this array at specified [indices].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sliceArray(indices:
Collection<Int>): UIntArray {\n return UIntArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array
containing elements of this array at specified [indices].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sliceArray(indices:
Collection<Int>): ULongArray {\n return ULongArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array
containing elements of this array at specified [indices].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sliceArray(indices:
Collection<Int>): UByteArray {\n return UByteArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array
containing elements of this array at specified [indices].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sliceArray(indices:
Collection<Int>): UShortArray {\n return UShortArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an
array containing elements at indices in the specified [indices] range.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sliceArray(indices: IntRange):
UIntArray {\n return UIntArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array containing elements at
indices in the specified [indices] range.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
ULongArray.sliceArray(indices: IntRange): ULongArray {\n return
ULongArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array containing elements at indices in the
specified [indices] range.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun

```

```

UByteArray.sliceArray(indices: IntRange): UByteArray {\n  return
UByteArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns an array containing elements at indices in the
specified [indices] range.\n * \n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
UShortArray.sliceArray(indices: IntRange): UShortArray {\n  return
UShortArray(storage.sliceArray(indices))\n}\n\n/**\n * Returns a list containing first [n] elements.\n * \n *
@throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.take(n: Int): List<UInt> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  if (n == 0) return emptyList()\n  if (n >=
size) return toList()\n  if (n == 1) return listOf(this[0])\n  var count = 0\n  val list = ArrayList<UInt>()\n  for
(item in this) {\n    if (count++ == n)\n      break\n    list.add(item)\n  }\n  return list\n}\n\n/**\n * Returns
a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.take(n: Int): List<ULong> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  if (n == 0) return emptyList()\n  if (n >=
size) return toList()\n  if (n == 1) return listOf(this[0])\n  var count = 0\n  val list = ArrayList<ULong>()\n  for
(item in this) {\n    if (count++ == n)\n      break\n    list.add(item)\n  }\n  return list\n}\n\n/**\n * Returns
a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.take(n: Int): List<UByte> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  if (n == 0) return emptyList()\n  if (n >=
size) return toList()\n  if (n == 1) return listOf(this[0])\n  var count = 0\n  val list = ArrayList<UByte>()\n  for
(item in this) {\n    if (count++ == n)\n      break\n    list.add(item)\n  }\n  return list\n}\n\n/**\n * Returns
a list containing first [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.take(n: Int): List<UShort> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  if (n == 0) return emptyList()\n  if (n >=
size) return toList()\n  if (n == 1) return listOf(this[0])\n  var count = 0\n  val list = ArrayList<UShort>()\n  for
(item in this) {\n    if (count++ == n)\n      break\n    list.add(item)\n  }\n  return list\n}\n\n/**\n * Returns
a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.takeLast(n: Int): List<UInt> {\n
require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  if (n == 0) return emptyList()\n  val size =
size\n  if (n >= size) return toList()\n  if (n == 1) return listOf(this[size - 1])\n  val list = ArrayList<UInt>()\n
for (index in size - n until size)\n    list.add(this[index])\n  return list\n}\n\n/**\n * Returns a list containing last
[n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.takeLast(n: Int): List<ULong>
{\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  if (n == 0) return emptyList()\n  val
size = size\n  if (n >= size) return toList()\n  if (n == 1) return listOf(this[size - 1])\n  val list =
ArrayList<ULong>()\n  for (index in size - n until size)\n    list.add(this[index])\n  return list\n}\n\n/**\n *
Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n *
@sample samples.collections.Collections.Transformations.take\n
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.takeLast(n: Int): List<UByte>
{\n  require(n >= 0) { \"Requested element count $n is less than zero.\" }\n  if (n == 0) return emptyList()\n  val
size = size\n  if (n >= size) return toList()\n  if (n == 1) return listOf(this[size - 1])\n  val list =
ArrayList<UByte>()\n  for (index in size - n until size)\n    list.add(this[index])\n  return list\n}\n\n/**\n *
Returns a list containing last [n] elements.\n * \n * @throws IllegalArgumentException if [n] is negative.\n * \n *
@sample samples.collections.Collections.Transformations.take\n

```

```

@sample samples.collections.Collections.Transformations.take\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.takeLast(n: Int): List<UShort>
{\n    require(n >= 0) { \"Requested element count $n is less than zero.\" }\n    if (n == 0) return emptyList()\n    val
size = size\n    if (n >= size) return toList()\n    if (n == 1) return listOf(this[size - 1])\n    val list =
ArrayList<UShort>(n)\n    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n\n/**\n *
Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.takeLastWhile(predicate: (UInt) -> Boolean): List<UInt> {\n    for (index in lastIndex downTo 0) {\n
if (!predicate(this[index])) {\n        return drop(index + 1)\n    }\n    }\n    return toList()\n}\n\n/**\n * Returns a
list containing last elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.takeLastWhile(predicate: (ULong) -> Boolean): List<ULong> {\n    for (index in lastIndex downTo 0)
{\n    if (!predicate(this[index])) {\n        return drop(index + 1)\n    }\n    }\n    return toList()\n}\n\n/**\n *
Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.takeLastWhile(predicate: (UByte) -> Boolean): List<UByte> {\n    for (index in lastIndex downTo 0)
{\n    if (!predicate(this[index])) {\n        return drop(index + 1)\n    }\n    }\n    return toList()\n}\n\n/**\n *
Returns a list containing last elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.takeLastWhile(predicate: (UShort) -> Boolean): List<UShort> {\n    for (index in lastIndex downTo 0)
{\n    if (!predicate(this[index])) {\n        return drop(index + 1)\n    }\n    }\n    return toList()\n}\n\n/**\n *
Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.takeWhile(predicate: (UInt) -> Boolean): List<UInt> {\n    val list = ArrayList<UInt>()\n    for (item in
this) {\n    if (!predicate(item))\n        break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n * Returns a list
containing first elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.takeWhile(predicate: (ULong) -> Boolean): List<ULong> {\n    val list = ArrayList<ULong>()\n    for
(item in this) {\n    if (!predicate(item))\n        break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n *
Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.takeWhile(predicate: (UByte) -> Boolean): List<UByte> {\n    val list = ArrayList<UByte>()\n    for
(item in this) {\n    if (!predicate(item))\n        break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n *
Returns a list containing first elements satisfying the given [predicate].\n * \n * @sample
samples.collections.Collections.Transformations.take\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.takeWhile(predicate: (UShort) -> Boolean): List<UShort> {\n    val list = ArrayList<UShort>()\n    for
(item in this) {\n    if (!predicate(item))\n        break\n        list.add(item)\n    }\n    return list\n}\n\n/**\n *
Reverses elements in the array in-place.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.reverse(): Unit {\n    storage.reverse()\n}\n\n/**\n * Reverses elements in the array in-place.\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.reverse(): Unit {\n    storage.reverse()\n}\n\n/**\n * Reverses elements in the array in-place.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.reverse(): Unit {\n    storage.reverse()\n}\n\n/**\n * Reverses elements in the array in-place.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.reverse(): Unit {\n    storage.reverse()\n}\n\n/**\n * Returns a list with elements in reversed order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.reversed(): List<UInt> {\n    if
(isEmpty()) return emptyList()\n    val list = toMutableList()\n    list.reverse()\n    return list\n}\n\n/**\n * Returns a
list with elements in reversed order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
ULongArray.reversed(): List<ULong> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n
list.reverse()\n    return list\n}\n\n/**\n * Returns a list with elements in reversed order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
UByteArray.reversed(): List<UByte> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n
list.reverse()\n    return list\n}\n\n/**\n * Returns
a list with elements in reversed order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
UShortArray.reversed(): List<UShort> {\n    if (isEmpty()) return emptyList()\n    val list = toMutableList()\n
list.reverse()\n    return list\n}\n\n/**\n * Returns an array with elements of this array in reversed order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.reversedArray(): UIntArray {\n    return UIntArray(storage.reversedArray())\n}\n\n/**\n * Returns an
array with elements of this array in reversed order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.reversedArray(): ULongArray {\n    return ULongArray(storage.reversedArray())\n}\n\n/**\n *
Returns an array with elements of this array in reversed order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.reversedArray(): UByteArray {\n    return UByteArray(storage.reversedArray())\n}\n\n/**\n * Returns
an array with elements of this array in reversed order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.reversedArray(): UShortArray {\n    return UShortArray(storage.reversedArray())\n}\n\n/**\n * Sorts
elements in the array in-place descending according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sortDescending(): Unit {\n    if
(size > 1) {\n        sort()\n        reverse()\n    }\n}\n\n/**\n * Sorts elements in the array in-place descending
according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
ULongArray.sortDescending(): Unit {\n    if (size > 1) {\n        sort()\n        reverse()\n    }\n}\n\n/**\n * Sorts
elements in the array in-place descending according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
UByteArray.sortDescending(): Unit {\n    if (size > 1) {\n        sort()\n        reverse()\n    }\n}\n\n/**\n * Sorts
elements in the array in-place descending according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
UShortArray.sortDescending(): Unit {\n    if (size > 1) {\n        sort()\n        reverse()\n    }\n}\n\n/**\n * Returns a
list of all elements sorted according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sorted(): List<UInt> {\n    return
copyOf().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all elements sorted according to their natural sort
order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
ULongArray.sorted(): List<ULong> {\n    return copyOf().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all
elements sorted according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
UByteArray.sorted(): List<UByte> {\n    return copyOf().apply { sort() }.asList()\n}\n\n/**\n * Returns a list of all
elements sorted according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
UShortArray.sorted(): List<UShort> {\n    return copyOf().apply { sort() }.asList()\n}\n\n/**\n * Returns an array with
all elements of this array sorted according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun

```

```

UIntArray.sortedArray(): UIntArray {n  if (isEmpty()) return this\n  return this.copyOf().apply { sort()
}\n}\n\n/**\n * Returns an array with all elements of this array sorted according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sortedArray(): ULongArray
{\n  if (isEmpty()) return this\n  return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all
elements of this array sorted according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sortedArray(): UByteArray {\n
  if (isEmpty()) return this\n  return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all elements
of this array sorted according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sortedArray(): UShortArray
{\n  if (isEmpty()) return this\n  return this.copyOf().apply { sort() }\n}\n\n/**\n * Returns an array with all
elements of this array sorted descending according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sortedArrayDescending():
UIntArray {\n  if (isEmpty()) return this\n  return this.copyOf().apply { sortDescending() }\n}\n\n/**\n * Returns
an array with all elements of this array sorted descending according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.sortedArrayDescending():
ULongArray {\n  if (isEmpty()) return this\n  return this.copyOf().apply { sortDescending() }\n}\n\n/**\n *
Returns an array with all elements of this array sorted descending according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sortedArrayDescending():
UByteArray {\n  if (isEmpty()) return this\n  return this.copyOf().apply { sortDescending() }\n}\n\n/**\n *
Returns an array with all elements of this array sorted descending according to their natural sort order.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.sortedArrayDescending():
UShortArray {\n  if (isEmpty()) return this\n  return this.copyOf().apply { sortDescending() }\n}\n\n/**\n *
Returns a list of all elements sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It
means that equal elements preserve their order relative to each other after sorting.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.sortedDescending(): List<UInt>
{\n  return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted descending
according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve their order
relative to each other after sorting.\n * \n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
ULongArray.sortedDescending(): List<ULong> {\n  return copyOf().apply { sort() }.reversed()\n}\n\n/**\n *
Returns a list of all elements sorted descending according to their natural sort order.\n * \n * The sort is _stable_. It
means that equal elements preserve their order relative to each other after sorting.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sortedDescending():
List<UByte> {\n  return copyOf().apply { sort() }.reversed()\n}\n\n/**\n * Returns a list of all elements sorted
descending according to their natural sort order.\n * \n * The sort is _stable_. It means that equal elements preserve
their order relative to each other after sorting.\n * \n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic
fun UShortArray.sortedDescending(): List<UShort> {\n  return copyOf().apply { sort() }.reversed()\n}\n\n/**\n *
Returns an array of type [ByteArray], which is a view of this array where each element is a signed reinterpretation\n
* of the corresponding element of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.asByteArray(): ByteArray {\n  return storage\n}\n\n/**\n * Returns an array of type [IntArray], which
is a view of this array where each element is a signed reinterpretation\n * of the corresponding element of this
array.\n * \n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.asIntArray(): IntArray {\n  return storage\n}\n\n/**\n * Returns a [List] that wraps the original array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun UIntArray.asList():
List<UInt>\n\n/**\n * Returns a [List] that wraps the original array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun ULongArray.asList():
List<ULong>\n\n/**\n * Returns a [List] that wraps the original array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun UByteArray.asList():

```

List<UByte>\n\n/**\n * Returns a [List] that wraps the original array.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun UShortArray.asList():
```

List<UShort>\n\n/**\n * Returns an array of type [LongArray], which is a view of this array where each element is a signed reinterpretation\n * of the corresponding element of this array.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

ULongArray.asLongArray(): LongArray {\n return storage}\n\n/**\n * Returns an array of type [ShortArray], which is a view of this array where each element is a signed reinterpretation\n * of the corresponding element of this array.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

UShortArray.asShortArray(): ShortArray {\n return storage}\n\n/**\n * Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

ByteArray.asUByteArray(): UByteArray {\n return UByteArray(this)}\n\n/**\n * Returns an array of type [UIntArray], which is a view of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

IntArray.asUIntArray(): UIntArray {\n return UIntArray(this)}\n\n/**\n * Returns an array of type [ULongArray], which is a view of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

LongArray.asULongArray(): ULongArray {\n return ULongArray(this)}\n\n/**\n * Returns an array of type [UShortArray], which is a view of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this array.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

ShortArray.asUShortArray(): UShortArray {\n return UShortArray(this)}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun
```

UIntArray.contentEquals(other: UIntArray): Boolean {\n return storage.contentEquals(other.storage)}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun
```

ULongArray.contentEquals(other: ULongArray): Boolean {\n return storage.contentEquals(other.storage)}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun
```

UByteArray.contentEquals(other: UByteArray): Boolean {\n return storage.contentEquals(other.storage)}\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun
```

UShortArray.contentEquals(other: UShortArray): Boolean {\n return storage.contentEquals(other.storage)}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
```

UIntArray.contentHashCode(): Int {\n return storage.contentHashCode()}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
```

ULongArray.contentHashCode(): Int {\n return storage.contentHashCode()}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
```

UByteArray.contentHashCode(): Int {\n return storage.contentHashCode()}\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
```

UShortArray.contentHashCode(): Int {\n return storage.contentHashCode()}\n\n/**\n * Returns a string representation of the contents of the specified array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n


```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.contentToString(): String {\n
return joinToString(", ", "[", "]")\n}\n\n/**\n * Returns a string representation of the contents of the specified
array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.contentToString(): String {\n
return joinToString(", ", "[", "]")\n}\n\n/**\n * Returns a string representation of the contents of the specified
array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.contentToString(): String {\n
return joinToString(", ", "[", "]")\n}\n\n/**\n * Returns a string representation of the contents of the specified
array as if it is [List].\n * \n * @sample samples.collections.Arrays.ContentOperations.contentToString\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.contentToString(): String {\n
return joinToString(", ", "[", "]")\n}\n\n/**\n * Copies this array or its subrange into the [destination] array and
returns that array.\n * \n * It's allowed to pass the same array in the [destination] and even specify the subrange so
that it overlaps with the destination range.\n * \n * @param destination the array to copy to.\n * @param
destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the
beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the
subrange to copy, size of this array by default.\n * \n * @throws IndexOutOfBoundsException or
[IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex
> endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array
starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices
range.\n * \n * @return the [destination] array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.copyInto(destination: UIntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
UIntArray {\n storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n return
destination}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n *
It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the
[destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy,
0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n *
@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the
subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index
is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.copyInto(destination: ULongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
ULongArray {\n storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n return
destination}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n *
It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the
[destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy,
0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n *
@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the
subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index
is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.copyInto(destination: UByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
UByteArray {\n storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n return
destination}\n\n/**\n * Copies this array or its subrange into the [destination] array and returns that array.\n * \n *
It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
destination range.\n * \n * @param destination the array to copy to.\n * @param destinationOffset the position in the
[destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy,
0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * \n *
@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the
subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index
is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n

```

It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
@param destination the array to copy to.
@param destinationOffset the position in the [destination] array to copy to, 0 by default.
@param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
@param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
or when that index is out of the [destination] array indices range.
@return the [destination] array.

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUShortArray.copyInto(destination: UShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int =\nsize): UShortArray {\n    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n    return\ndestination\n}\n\n/**\n * Returns new array which is a copy of the original array.\n *\n * @sample\n samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUIntArray.copyOf(): UIntArray {\n    return UIntArray(storage.copyOf())\n}\n\n/**\n * Returns new array which is\n a copy of the original array.\n *\n * @sample\n samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nULongArray.copyOf(): ULongArray {\n    return ULongArray(storage.copyOf())\n}\n\n/**\n * Returns new array\n which is a copy of the original array.\n *\n * @sample\n samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUByteArray.copyOf(): UByteArray {\n    return UByteArray(storage.copyOf())\n}\n\n/**\n * Returns new array\n which is a copy of the original array.\n *\n * @sample\n samples.collections.Arrays.CopyOfOperations.copyOfOf\n\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUShortArray.copyOf(): UShortArray {\n    return UShortArray(storage.copyOf())\n}\n\n/**\n * Returns new array\n which is a copy of the original array, resized to the given [newSize].\n * The copy is either truncated or padded at\n the end with zero values if necessary.\n *\n * - If [newSize] is less than the size of the original array, the copy array\n is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the\n copy array are filled with zero values.\n\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUIntArray.copyOf(newSize: Int): UIntArray {\n    return UIntArray(storage.copyOf(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either\n truncated or padded at the end with zero values if necessary.\n *\n * - If [newSize] is less than the size of the\n original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original\n array, the extra elements in the copy array are filled with zero values.\n\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nULongArray.copyOf(newSize: Int): ULongArray {\n    return ULongArray(storage.copyOf(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either\n truncated or padded at the end with zero values if necessary.\n *\n * - If [newSize] is less than the size of the\n original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original\n array, the extra elements in the copy array are filled with zero values.\n\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUByteArray.copyOf(newSize: Int): UByteArray {\n    return UByteArray(storage.copyOf(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either\n truncated or padded at the end with zero values if necessary.\n *\n * - If [newSize] is less than the size of the\n original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original\n array, the extra elements in the copy array are filled with zero values.\n\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun\nUShortArray.copyOf(newSize: Int): UShortArray {\n    return UShortArray(storage.copyOf(newSize))\n}\n\n/**\n * Returns new array which is a copy of the original array, resized to the given [newSize].\n * The copy is either\n truncated or padded at the end with zero values if necessary.\n *\n * - If [newSize] is less than the size of the\n original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original\n array, the extra elements in the copy array are filled with zero values.\n\n*/
```

Returns a new array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun  
UIntArray.copyOfRange(fromIndex: Int, toIndex: Int): UIntArray {\n    return
```

```
UIntArray(storage.copyOfRange(fromIndex, toIndex))\n}\n\n/**\n * Returns a new array which is a copy of the  
specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in  
`0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

```
ULongArray.copyOfRange(fromIndex: Int, toIndex: Int): ULongArray {\n    return  
ULongArray(storage.copyOfRange(fromIndex, toIndex))\n}\n\n/**\n * Returns a new array which is a copy of the  
specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in  
`0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

```
UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {\n    return  
UByteArray(storage.copyOfRange(fromIndex, toIndex))\n}\n\n/**\n * Returns a new array which is a copy of the  
specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in  
`0..array.size`\n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`\n
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

```
UShortArray.copyOfRange(fromIndex: Int, toIndex: Int): UShortArray {\n    return  
UShortArray(storage.copyOfRange(fromIndex, toIndex))\n}\n\n/**\n * Fills this array or its subrange with the  
specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive), 0 by default.\n * @param  
toIndex the end of the range (exclusive), size of this array by default.\n * \n * @throws
```

```
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *  
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.fill(element: UInt, fromIndex:  
Int = 0, toIndex: Int = size): Unit {\n    storage.fill(element.toInt(), fromIndex, toIndex)\n}\n\n/**\n * Fills this array  
or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range (inclusive), 0 by  
default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n * @throws  
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *  
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.fill(element: ULong,  
fromIndex: Int = 0, toIndex: Int = size): Unit {\n    storage.fill(element.toLong(), fromIndex, toIndex)\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range  
(inclusive), 0 by default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n *  
@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this  
array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.fill(element: UByte,  
fromIndex: Int = 0, toIndex: Int = size): Unit {\n    storage.fill(element.toByte(), fromIndex, toIndex)\n}\n\n/**\n * Fills this array or its subrange with the specified [element] value.\n * \n * @param fromIndex the start of the range  
(inclusive), 0 by default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * \n *  
@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this  
array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.fill(element: UShort,  
fromIndex: Int = 0, toIndex: Int = size): Unit {\n    storage.fill(element.toShort(), fromIndex, toIndex)\n}\n\n/**\n * Returns the range of valid indices for the array.\n
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline val UIntArray.indices: IntRange\n    get()  
= storage.indices\n\n/**\n * Returns the range of valid indices for the array.\n
```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline val ULongArray.indices: IntRange\n
get() = storage.indices\n\n/**\n * Returns the range of valid indices for the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline val UByteArray.indices: IntRange\n
get() = storage.indices\n\n/**\n * Returns the range of valid indices for the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline val UShortArray.indices: IntRange\n
get() = storage.indices\n\n/**\n * Returns the last valid index for the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline val UIntArray.lastIndex: Int\n get() =
storage.lastIndex\n\n/**\n * Returns the last valid index for the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline val ULongArray.lastIndex: Int\n get() =
storage.lastIndex\n\n/**\n * Returns the last valid index for the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline val UByteArray.lastIndex: Int\n get() =
storage.lastIndex\n\n/**\n * Returns the last valid index for the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline val UShortArray.lastIndex: Int\n get() =
storage.lastIndex\n\n/**\n * Returns an array containing all elements of the original array and then the given
[element].\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline
operator fun UIntArray.plus(element: UInt): UIntArray {\n return UIntArray(storage +
element.toInt())\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given
[element].\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline
operator fun ULongArray.plus(element: ULong): ULongArray {\n return ULongArray(storage +
element.toLong())\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given
[element].\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline
operator fun UByteArray.plus(element: UByte): UByteArray {\n return UByteArray(storage +
element.toByte())\n}\n\n/**\n * Returns an array containing all elements of the original array and then the given
[element].\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline
operator fun UShortArray.plus(element: UShort): UShortArray {\n return UShortArray(storage +
element.toShort())\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements
of the given [elements] collection.\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic operator
fun UIntArray.plus(elements: Collection<UInt>): UIntArray {\n var index = size\n val result =
storage.copyOfOf(size + elements.size)\n for (element in elements) result[index++] = element.toInt()\n return
UIntArray(result)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements
of the given [elements] collection.\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic operator
fun ULongArray.plus(elements: Collection<ULong>): ULongArray {\n var index = size\n val result =
storage.copyOfOf(size + elements.size)\n for (element in elements) result[index++] = element.toLong()\n return
ULongArray(result)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all
elements of the given [elements] collection.\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic
operator fun UByteArray.plus(elements: Collection<UByte>): UByteArray {\n var index = size\n val result =
storage.copyOfOf(size + elements.size)\n for (element in elements) result[index++] = element.toByte()\n return
UByteArray(result)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements
of the given [elements] collection.\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic operator
fun UShortArray.plus(elements: Collection<UShort>): UShortArray {\n var index = size\n val result =
storage.copyOfOf(size + elements.size)\n for (element in elements) result[index++] = element.toShort()\n return
UShortArray(result)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all
elements of the given [elements] array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun
UIntArray.plus(elements: UIntArray): UIntArray {\n return UIntArray(storage + elements.storage)\n}\n\n/**\n *
Returns an array containing all elements of the original array and then all elements of the given [elements] array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun
ULongArray.plus(elements: ULongArray): ULongArray {\n return ULongArray(storage +

```

```

elements.storage)\n\n/**\n * Returns an array containing all elements of the original array and then all elements
of the given [elements] array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun
UByteArray.plus(elements: UByteArray): UByteArray {\n    return UByteArray(storage +
elements.storage)\n}\n\n/**\n * Returns an array containing all elements of the original array and then all elements
of the given [elements] array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun
UShortArray.plus(elements: UShortArray): UShortArray {\n    return UShortArray(storage +
elements.storage)\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic
fun UIntArray.sort(): Unit {\n    if (size > 1) sortArray(this)\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic
fun ULongArray.sort(): Unit {\n    if (size > 1) sortArray(this)\n}\n\n/**\n * Sorts the array in-place.\n * \n *
@sample samples.collections.Arrays.Sorting.sortArray\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.sort(): Unit {\n    if (size > 1)
sortArray(this)\n}\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic
fun UShortArray.sort(): Unit {\n    if (size > 1) sortArray(this)\n}\n\n/**\n * Returns an array of type [ByteArray],
which is a copy of this array where each element is a signed reinterpretation\n * of the corresponding element of this
array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.toByteArray(): ByteArray {\n    return storage.copyOf()\n}\n\n/**\n * Returns an array of type
[IntArray], which is a copy of this array where each element is a signed reinterpretation\n * of the corresponding
element of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.toIntArray(): IntArray {\n    return storage.copyOf()\n}\n\n/**\n * Returns an array of type [LongArray],
which is a copy of this array where each element is a signed reinterpretation\n * of the corresponding element of this
array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.toLongArray(): LongArray {\n    return storage.copyOf()\n}\n\n/**\n * Returns an array of type
[ShortArray], which is a copy of this array where each element is a signed reinterpretation\n * of the corresponding
element of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.toShortArray(): ShortArray {\n    return storage.copyOf()\n}\n\n/**\n * Returns a *typed* object array
containing all of the elements of this primitive array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.toTypedArray(): Array<UInt>
{\n    return Array(size) { index -> this[index] }\n}\n\n/**\n * Returns a *typed* object array containing all of the
elements of this primitive array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
ULongArray.toTypedArray(): Array<ULong> {\n    return Array(size) { index -> this[index] }\n}\n\n/**\n *
Returns a *typed* object array containing all of the elements of this primitive array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.toTypedArray():
Array<UByte> {\n    return Array(size) { index -> this[index] }\n}\n\n/**\n * Returns a *typed* object array
containing all of the elements of this primitive array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.toTypedArray():
Array<UShort> {\n    return Array(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of UByte containing
all of the elements of this generic array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Array<out UByte>.toUByteArray(): UByteArray {\n    return UByteArray(size) { index -> this[index] }\n}\n\n/**\n *
Returns an array of type [UByteArray], which is a copy of this array where each element is an unsigned
reinterpretation\n * of the corresponding element of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

```

```

ByteArray.toByteArray(): UByteArray {\n  return UByteArray(this.copyOf())\n}\n\n/**\n * Returns an array of
UInt containing all of the elements of this generic array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun Array<out UInt>.toUIntArray(): UIntArray
{\n  return UIntArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of type [UIntArray], which is a
copy of this array where each element is an unsigned reinterpretation\n * of the corresponding element of this
array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
IntArray.toUIntArray(): UIntArray {\n  return UIntArray(this.copyOf())\n}\n\n/**\n * Returns an array of ULong
containing all of the elements of this generic array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun Array<out ULong>.toULongArray():
ULongArray {\n  return ULongArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of type
[ULongArray], which is a copy of this array where each element is an unsigned reinterpretation\n * of the
corresponding element of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
LongArray.toULongArray(): ULongArray {\n  return ULongArray(this.copyOf())\n}\n\n/**\n * Returns an array
of UShort containing all of the elements of this generic array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun Array<out UShort>.toUShortArray():
UShortArray {\n  return UShortArray(size) { index -> this[index] }\n}\n\n/**\n * Returns an array of type
[UShortArray], which is a copy of this array where each element is an unsigned reinterpretation\n * of the
corresponding element of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ShortArray.toUShortArray(): UShortArray {\n  return UShortArray(this.copyOf())\n}\n\n/**\n * Returns a single
list of all elements yielded from results of [transform] function being invoked on each element of original array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
UIntArray.flatMap(transform: (UInt) -> Iterable<R>): List<R> {\n  return flatMapTo(ArrayList<R>(),
transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being
invoked on each element of original array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
ULongArray.flatMap(transform: (ULong) -> Iterable<R>): List<R> {\n  return flatMapTo(ArrayList<R>(),
transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being
invoked on each element of original array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
UByteArray.flatMap(transform: (UByte) -> Iterable<R>): List<R> {\n  return flatMapTo(ArrayList<R>(),
transform)\n}\n\n/**\n * Returns a single list of all elements yielded from results of [transform] function being
invoked on each element of original array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
UShortArray.flatMap(transform: (UShort) -> Iterable<R>): List<R> {\n  return flatMapTo(ArrayList<R>(),
transform)\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each
element of original array, to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UIntArray.flatMapTo(destination: C, transform: (UInt) -> Iterable<R>): C {\n  for
(element in this) {\n    val list = transform(element)\n    destination.addAll(list)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each
element of original array, to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> ULongArray.flatMapTo(destination: C, transform: (ULong) -> Iterable<R>): C {\n  for
(element in this) {\n    val list = transform(element)\n    destination.addAll(list)\n  }\n  return
destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each
element of original array, to the given [destination].\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UByteArray.flatMapTo(destination: C, transform: (UByte) -> Iterable<R>): C {\n for
(element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return
destination\n}\n\n/**\n * Appends all elements yielded from results of [transform] function being invoked on each
element of original array, to the given [destination].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UShortArray.flatMapTo(destination: C, transform: (UShort) -> Iterable<R>): C {\n for
(element in this) {\n val list = transform(element)\n destination.addAll(list)\n }\n return
destination\n}\n\n/**\n * Groups elements of the original array by the key returned by the given [keySelector]
function\n * applied to each element and returns a map where each group key is associated with a list of
corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced from the
original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K>
UIntArray.groupBy(keySelector: (UInt) -> K): Map<K, List<UInt>> {\n return groupByTo(LinkedHashMap<K,
MutableList<UInt>>(), keySelector)\n}\n\n/**\n * Groups elements of the original array by the key returned by the
given [keySelector] function\n * applied to each element and returns a map where each group key is associated with
a list of corresponding elements.\n * \n * The returned map preserves the entry iteration order of the keys produced
from the original array.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K>
ULongArray.groupBy(keySelector: (ULong) -> K): Map<K, List<ULong>> {\n return
groupByTo(LinkedHashMap<K, MutableList<ULong>>(), keySelector)\n}\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map
where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the
entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupBy\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K>
UByteArray.groupBy(keySelector: (UByte) -> K): Map<K, List<UByte>> {\n return
groupByTo(LinkedHashMap<K, MutableList<UByte>>(), keySelector)\n}\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map
where each group key is associated with a list of corresponding elements.\n * \n * The returned map preserves the
entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupBy\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K>
UShortArray.groupBy(keySelector: (UShort) -> K): Map<K, List<UShort>> {\n return
groupByTo(LinkedHashMap<K, MutableList<UShort>>(), keySelector)\n}\n\n/**\n * Groups values returned by
the [valueTransform] function applied to each element of the original array\n * by the key returned by the given
[keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of
corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the
original array.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V>
UIntArray.groupBy(keySelector: (UInt) -> K, valueTransform: (UInt) -> V): Map<K, List<V>> {\n return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values
returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by
the given [keySelector] function applied to the element\n * and returns a map where each group key is associated
with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V>

```

```

ULongArray.groupBy(keySelector: (ULong) -> K, valueTransform: (ULong) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values
returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by
the given [keySelector] function applied to the element\n * and returns a map where each group key is associated
with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V>
UByteArray.groupBy(keySelector: (UByte) -> K, valueTransform: (UByte) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups values
returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by
the given [keySelector] function applied to the element\n * and returns a map where each group key is associated
with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V>
UShortArray.groupBy(keySelector: (UShort) -> K, valueTransform: (UShort) -> V): Map<K, List<V>> {\n  return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n/**\n * Groups elements
of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to
the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The
[destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, M :
MutableMap<in K, MutableList<UInt>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K): M {\n
for (element in this) {\n    val key = keySelector(element)\n    val list = destination.getOrPut(key) {\n
ArrayList<UInt>()\n    list.add(element)\n    }\n    return destination\n}\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the
[destination] map each group key associated with a list of corresponding elements.\n * \n * @return The
[destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, M :
MutableMap<in K, MutableList<ULong>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K):
M {\n for (element in this) {\n    val key = keySelector(element)\n    val list = destination.getOrPut(key) {\n
ArrayList<ULong>()\n    list.add(element)\n    }\n    return destination\n}\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the
[destination] map each group key associated with a list of corresponding elements.\n * \n * @return The
[destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, M :
MutableMap<in K, MutableList<UByte>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K):
M {\n for (element in this) {\n    val key = keySelector(element)\n    val list = destination.getOrPut(key) {\n
ArrayList<UByte>()\n    list.add(element)\n    }\n    return destination\n}\n\n/**\n * Groups elements of the
original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the
[destination] map each group key associated with a list of corresponding elements.\n * \n * @return The
[destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, M :
MutableMap<in K, MutableList<UShort>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K):
M {\n for (element in this) {\n    val key = keySelector(element)\n    val list = destination.getOrPut(key) {\n
ArrayList<UShort>()\n    list.add(element)\n    }\n    return destination\n}\n\n/**\n * Groups values returned by
the [valueTransform] function applied to each element of the original array\n * by the key returned by the given
[keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a

```



```

list of corresponding values.\n * \n * @return The [destination] map.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeyAndValues\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V,
M : MutableMap<in K, MutableList<V>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K,
valueTransform: (UInt) -> V): M {\n  for (element in this) {\n    val key = keySelector(element)\n    val list =
destination.getOrPut(key) { ArrayList<V>() }\n    list.add(valueTransform(element))\n  }\n  return
destination\n}\n\n**\n * Groups values returned by the [valueTransform] function applied to each element of the
original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the
[destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]
map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeyAndValues\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V,
M : MutableMap<in K, MutableList<V>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K,
valueTransform: (ULong) -> V): M {\n  for (element in this) {\n    val key = keySelector(element)\n    val list =
destination.getOrPut(key) { ArrayList<V>() }\n    list.add(valueTransform(element))\n  }\n  return
destination\n}\n\n**\n * Groups values returned by the [valueTransform] function applied to each element of the
original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the
[destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]
map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeyAndValues\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V,
M : MutableMap<in K, MutableList<V>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K,
valueTransform: (UByte) -> V): M {\n  for (element in this) {\n    val key = keySelector(element)\n    val list =
destination.getOrPut(key) { ArrayList<V>() }\n    list.add(valueTransform(element))\n  }\n  return
destination\n}\n\n**\n * Groups values returned by the [valueTransform] function applied to each element of the
original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the
[destination] map each group key associated with a list of corresponding values.\n * \n * @return The [destination]
map.\n * \n * @sample samples.collections.Collections.Transformations.groupByKeyAndValues\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V,
M : MutableMap<in K, MutableList<V>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K,
valueTransform: (UShort) -> V): M {\n  for (element in this) {\n    val key = keySelector(element)\n    val list =
destination.getOrPut(key) { ArrayList<V>() }\n    list.add(valueTransform(element))\n  }\n  return
destination\n}\n\n**\n * Returns a list containing the results of applying the given [transform] function\n * to each
element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
UIntArray.map(transform: (UInt) -> R): List<R> {\n  return mapTo(ArrayList<R>(size), transform)\n}\n\n**\n * Returns a list containing the results of applying the given [transform] function\n * to each element in the original
array.\n * \n * @sample samples.collections.Collections.Transformations.map\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
ULongArray.map(transform: (ULong) -> R): List<R> {\n  return mapTo(ArrayList<R>(size),
transform)\n}\n\n**\n * Returns a list containing the results of applying the given [transform] function\n * to each
element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
UByteArray.map(transform: (UByte) -> R): List<R> {\n  return mapTo(ArrayList<R>(size),
transform)\n}\n\n**\n * Returns a list containing the results of applying the given [transform] function\n * to each
element in the original array.\n * \n * @sample samples.collections.Collections.Transformations.map\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
UShortArray.map(transform: (UShort) -> R): List<R> {\n  return mapTo(ArrayList<R>(size),
transform)\n}\n\n**\n * Returns a list containing the results of applying the given [transform] function\n * to each
element and its index in the original array.\n * @param [transform] function that takes the index of an element and

```

the element itself\n * and returns the result of the transform applied to the element.\n

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
UIntArray.mapIndexed(transform: (index: Int, UInt) -> R): List<R> {\n    return
mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the
given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function
that takes the index of an element and the element itself\n * and returns the result of the transform applied to the
element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline
fun <R> ULongArray.mapIndexed(transform: (index: Int, ULong) -> R): List<R> {\n    return
mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the
given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function
that takes the index of an element and the element itself\n * and returns the result of the transform applied to the
element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline
fun <R> UByteArray.mapIndexed(transform: (index: Int, UByte) -> R): List<R> {\n    return
mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Returns a list containing the results of applying the
given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function
that takes the index of an element and the element itself\n * and returns the result of the transform applied to the
element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline
fun <R> UShortArray.mapIndexed(transform: (index: Int, UShort) -> R): List<R> {\n    return
mapIndexedTo(ArrayList<R>(size), transform)\n}\n\n/**\n * Applies the given [transform] function to each
element and its index in the original array\n * and appends the results to the given [destination].\n * @param
[transform] function that takes the index of an element and the element itself\n * and returns the result of the
transform applied to the element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UIntArray.mapIndexedTo(destination: C, transform: (index: Int, UInt) -> R): C {\n    var
index = 0\n    for (item in this)\n        destination.add(transform(index++, item))\n    return destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given [destination].\n * @param [transform] function that takes the index of an element and the element
itself\n * and returns the result of the transform applied to the element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> ULongArray.mapIndexedTo(destination: C, transform: (index: Int, ULong) -> R): C {\n    var
index = 0\n    for (item in this)\n        destination.add(transform(index++, item))\n    return
destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n *
and appends the results to the given [destination].\n * @param [transform] function that takes the index of an
element and the element itself\n * and returns the result of the transform applied to the element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UByteArray.mapIndexedTo(destination: C, transform: (index: Int, UByte) -> R): C {\n    var
index = 0\n    for (item in this)\n        destination.add(transform(index++, item))\n    return
destination\n}\n\n/**\n * Applies the given [transform] function to each element and its index in the original array\n *
and appends the results to the given [destination].\n * @param [transform] function that takes the index of an
element and the element itself\n * and returns the result of the transform applied to the element.\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UShortArray.mapIndexedTo(destination: C, transform: (index: Int, UShort) -> R): C {\n    var
index = 0\n    for (item in this)\n        destination.add(transform(index++, item))\n    return
destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and
appends the results to the given [destination].\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :
MutableCollection<in R>> UIntArray.mapTo(destination: C, transform: (UInt) -> R): C {\n    for (item in this)\n        destination.add(transform(item))\n    return destination\n}\n\n/**\n * Applies the given [transform] function to each

```

element of the original array\n * and appends the results to the given [destination].\n

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> ULongArray.mapTo(destination: C, transform: (ULong) -> R): C {\n    for (item in this)\n        destination.add(transform(item))\n    return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UByteArray.mapTo(destination: C, transform: (UByte) -> R): C {\n    for (item in this)\n        destination.add(transform(item))\n    return destination\n}\n\n/**\n * Applies the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UShortArray.mapTo(destination: C, transform: (UShort) -> R): C {\n    for (item in this)\n        destination.add(transform(item))\n    return destination\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.withIndex(): Iterable<IndexedValue<UInt>> {\n    return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.withIndex(): Iterable<IndexedValue<ULong>> {\n    return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.withIndex(): Iterable<IndexedValue<UByte>> {\n    return IndexingIterable { iterator() }\n}\n\n/**\n * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.withIndex(): Iterable<IndexedValue<UShort>> {\n    return IndexingIterable { iterator() }\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n */\n*\n@sample samples.collections.Collections.Aggregates.all\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.all(predicate: (UInt) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n */\n*\n@sample samples.collections.Collections.Aggregates.all\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.all(predicate: (ULong) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n */\n*\n@sample samples.collections.Collections.Aggregates.all\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.all(predicate: (UByte) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}\n\n/**\n * Returns `true` if all elements match the given [predicate].\n */\n*\n@sample samples.collections.Collections.Aggregates.all\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.all(predicate: (UShort) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}\n\n/**\n * Returns `true` if array has at least one element.\n */\n*\n@sample samples.collections.Collections.Aggregates.any\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.any(): Boolean {\n    return storage.any()\n}\n\n/**\n * Returns `true` if array has at least one element.\n */\n*\n@sample samples.collections.Collections.Aggregates.any\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.any(): Boolean {\n    return storage.any()\n}\n\n/**\n * Returns `true` if array has at least one element.\n */\n*\n@sample samples.collections.Collections.Aggregates.any\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
```

```

UByteArray.any(): Boolean {\n  return storage.any()\n}\n\n/**\n * Returns `true` if array has at least one
element.\n * \n * @sample samples.collections.Collections.Aggregates.any\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.any(): Boolean {\n  return storage.any()\n}\n\n/**\n * Returns `true` if at least one element matches
the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.any(predicate: (UInt) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return true\n
return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n * @sample
samples.collections.Collections.Aggregates.anyWithPredicate\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.any(predicate: (ULong) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return
true\n  return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n *
@sample samples.collections.Collections.Aggregates.anyWithPredicate\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.any(predicate: (UByte) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return
true\n  return false\n}\n\n/**\n * Returns `true` if at least one element matches the given [predicate].\n * \n *
@sample samples.collections.Collections.Aggregates.anyWithPredicate\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.any(predicate: (UShort) -> Boolean): Boolean {\n  for (element in this) if (predicate(element)) return
true\n  return false\n}\n\n/**\n * Returns the number of elements matching the given [predicate].\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.count(predicate: (UInt) -> Boolean): Int {\n  var count = 0\n  for (element in this) if
(predicate(element)) ++count\n  return count\n}\n\n/**\n * Returns the number of elements matching the given
[predicate].\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic
inline fun ULongArray.count(predicate: (ULong) -> Boolean): Int {\n  var count = 0\n  for (element in this) if
(predicate(element)) ++count\n  return count\n}\n\n/**\n * Returns the number of elements matching the given
[predicate].\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic
inline fun UByteArray.count(predicate: (UByte) -> Boolean): Int {\n  var count = 0\n  for (element in this) if
(predicate(element)) ++count\n  return count\n}\n\n/**\n * Returns the number of elements matching the given
[predicate].\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic
inline fun UShortArray.count(predicate: (UShort) -> Boolean): Int {\n  var count = 0\n  for (element in this) if
(predicate(element)) ++count\n  return count\n}\n\n/**\n * Accumulates value starting with [initial] value and
applying [operation] from left to right to current accumulator value and each element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
UIntArray.fold(initial: R, operation: (acc: R, UInt) -> R): R {\n  var accumulator = initial\n  for (element in
this) accumulator = operation(accumulator, element)\n  return accumulator\n}\n\n/**\n * Accumulates value starting
with [initial] value and applying [operation] from left to right to current accumulator value and each element.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>
ULongArray.fold(initial: R, operation: (acc: R, ULong) -> R): R {\n  var accumulator = initial\n  for (element in
this) accumulator = operation(accumulator, element)\n  return accumulator\n}\n\n/**\n * Accumulates value
starting with [initial] value and applying [operation] from left to right to current accumulator value and each
element.\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline
fun <R> UByteArray.fold(initial: R, operation: (acc: R, UByte) -> R): R {\n  var accumulator = initial\n  for
(element in this) accumulator = operation(accumulator, element)\n  return accumulator\n}\n\n/**\n * Accumulates
value starting with [initial] value and applying [operation] from left to right to current accumulator value and each
element.\n *\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline
fun <R> UShortArray.fold(initial: R, operation: (acc: R, UShort) -> R): R {\n  var accumulator = initial\n  for
(element in this) accumulator = operation(accumulator, element)\n  return accumulator\n}\n\n/**\n * Accumulates

```

value starting with [initial] value and applying [operation] from left to right
* to current accumulator value and each element with its index in the original array.
* @param [operation] function that takes the index of an element, current accumulator value
* and the element itself, and calculates the next accumulator value.

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nUIntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): R {\n    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to\n * right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation]\n * function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nULongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): R {\n    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to\n * right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation]\n * function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nUByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {\n    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from left to\n * right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation]\n * function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nUShortArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UShort) -> R): R {\n    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nUIntArray.foldRight(initial: R, operation: (UInt, acc: R) -> R): R {\n    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {\n        accumulator = operation(get(index--), accumulator)\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right to left\n * to each element and current accumulator value.
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nULongArray.foldRight(initial: R, operation: (ULong, acc: R) -> R): R {\n    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {\n        accumulator = operation(get(index--), accumulator)\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right\n * to left to each element and current accumulator value.
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nUByteArray.foldRight(initial: R, operation: (UByte, acc: R) -> R): R {\n    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {\n        accumulator = operation(get(index--), accumulator)\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right\n * to left to each element and current accumulator value.
```

```
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R>\nUShortArray.foldRight(initial: R, operation: (UShort, acc: R) -> R): R {\n    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {\n        accumulator = operation(get(index--), accumulator)\n    }\n    return accumulator\n}\n\n/**\n * Accumulates value starting with [initial] value and applying [operation] from right
```

to left to each element with its index in the original array and current accumulator value. @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

```

* Since Kotlin("1.3") ExperimentalUnsignedTypes @kotlin.internal.InlineOnly public inline fun <R>
UIntArray.foldRightIndexed(initial: R, operation: (index: Int, UInt, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element with its index in the original array and current accumulator value. @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

```

* Since Kotlin("1.3") ExperimentalUnsignedTypes @kotlin.internal.InlineOnly public inline fun <R>
ULongArray.foldRightIndexed(initial: R, operation: (index: Int, ULong, acc: R) -> R): R {
    var index =
    lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index),
        accumulator)
        --index
    }
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element with its index in the original array and current accumulator value. @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

```

* Since Kotlin("1.3") ExperimentalUnsignedTypes @kotlin.internal.InlineOnly public inline fun <R>
UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index =
    lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index),
        accumulator)
        --index
    }
    return accumulator
}

```

Accumulates value starting with [initial] value and applying [operation] from right to left to each element with its index in the original array and current accumulator value. @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

```

* Since Kotlin("1.3") ExperimentalUnsignedTypes @kotlin.internal.InlineOnly public inline fun <R>
UShortArray.foldRightIndexed(initial: R, operation: (index: Int, UShort, acc: R) -> R): R {
    var index =
    lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index),
        accumulator)
        --index
    }
    return accumulator
}

```

Performs the given [action] on each element.

```

* Since Kotlin("1.3") ExperimentalUnsignedTypes @kotlin.internal.InlineOnly public inline
fun UIntArray.forEach(action: (UInt) -> Unit): Unit {
    for (element in this) action(element)
}

```

Performs the given [action] on each element.

```

* Since Kotlin("1.3") ExperimentalUnsignedTypes @kotlin.internal.InlineOnly public inline fun
ULongArray.forEach(action: (ULong) -> Unit): Unit {
    for (element in this) action(element)
}

```

Performs the given [action] on each element.

```

* Since Kotlin("1.3") ExperimentalUnsignedTypes @kotlin.internal.InlineOnly public inline fun
UByteArray.forEach(action: (UByte) -> Unit): Unit {
    for (element in this) action(element)
}

```

Performs the given [action] on each element.

```

* Since Kotlin("1.3") ExperimentalUnsignedTypes @kotlin.internal.InlineOnly public inline fun
UShortArray.forEach(action: (UShort) -> Unit): Unit {
    for (element in this) action(element)
}

```

Performs the given [action] on each element, providing sequential index with the element. @param [action] function that takes the index of an element and the element itself and performs the desired action on the element.

```

* Since Kotlin("1.3") ExperimentalUnsignedTypes @kotlin.internal.InlineOnly public inline
fun UIntArray.forEachIndexed(action: (index: Int, UInt) -> Unit): Unit {
    var index = 0
    for (item in this)
    action(index++, item)
}

```

Performs the given [action] on each element, providing sequential index with the element. @param [action] function that takes the index of an element and the element itself and performs the desired action on the element.

```

* Since Kotlin("1.3") ExperimentalUnsignedTypes @kotlin.internal.InlineOnly public inline fun
ULongArray.forEachIndexed(action: (index: Int, ULong) -> Unit): Unit {
    var index = 0
    for (item in this)

```

action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the desired action on the element.\n

```
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun  
UByteArray.forEachIndexed(action: (index: Int, UByte) -> Unit): Unit {\n    var index = 0\n    for (item in this)  
        action(index++, item)\n}\n\n/**\n * Performs the given [action] on each element, providing sequential index with the element.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the desired action on the element.\n
```

```
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun  
UShortArray.forEachIndexed(action: (index: Int, UShort) -> Unit): Unit {\n    var index = 0\n    for (item in this)  
        action(index++, item)\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n
```

```
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.max(): UInt? {\n    if (isEmpty())  
        return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n
```

```
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.max(): ULong? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n
```

```
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.max(): UByte? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the largest element or `null` if there are no elements.\n
```

```
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.max(): UShort? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (max < e) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxBy\n
```

```
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UIntArray.maxBy(selector: (UInt) -> R): UInt? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxBy\n
```

```
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ULongArray.maxBy(selector: (ULong) -> R): ULong? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxBy\n
```

```
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UByteArray.maxBy(selector: (UByte) -> R): UByte? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}\n\n/**\n * Returns the first element yielding the largest value of the given function or `null` if there are no elements.\n * \n * @sample samples.collections.Collections.Aggregates.maxBy\n
```

```
*\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.maxBy(selector: (UShort) -> R): UShort? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue =
```

```

selector(maxElem)\n for (i in 1..lastIndex) {\n     val e = this[i]\n     val v = selector(e)\n     if (maxValue < v)\n     {\n         maxElem = e\n         maxValue = v\n     }\n }\n return maxElem\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.maxWith(comparator: Comparator<in UInt>): UInt? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.maxWith(comparator: Comparator<in ULong>): ULong? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.maxWith(comparator: Comparator<in UByte>): UByte? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.maxWith(comparator: Comparator<in UShort>): UShort? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e\n    }\n    return max\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.min(): UInt? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.min(): ULong? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.min(): UByte? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the smallest element or `null` if there are no elements.\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.min(): UShort? {\n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (min > e) min = e\n    }\n    return min\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n */\n\n@sample\nsamples.collections.Collections.Aggregates.minBy\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UIntArray.minBy(selector: (UInt) -> R): UInt? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v)\n        {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n */\n\n@sample\nsamples.collections.Collections.Aggregates.minBy\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ULongArray.minBy(selector: (ULong) -> R): ULong? {\n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v)\n        {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n\n/**\n * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.\n */\n\n@sample\nsamples.collections.Collections.Aggregates.minBy\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UByteArray.minBy(selector: (UByte) -> R): UByte? {\n    if (isEmpty()) return null\n    var

```



```

ByteArray.none(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
Returns `true` if no elements match the given [predicate].
@sample
samples.collections.Collections.Aggregates.noneWithPredicate

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
UShortArray.none(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
UIntArray.reduce(operation: (acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
ULongArray.reduce(operation: (acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
ByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
UShortArray.reduce(operation: (acc: UShort, UShort) -> UShort): UShort {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.
@param [operation] function that takes the index of an element, current accumulator value and the element itself and calculates the next accumulator value.

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
UIntArray.reduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.
@param [operation] function that takes the index of an element, current accumulator value and the element itself and calculates the next accumulator value.

@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
ULongArray.reduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.
@param [operation] function that takes the index of an element, current accumulator value and the element itself and calculates the next accumulator value.

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.reduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte {\n  if (isEmpty())\n  throw UnsupportedOperationException("Empty array can't be reduced.")\n  var accumulator = this[0]\n  for
(index in 1..lastIndex) {\n    accumulator = operation(index, accumulator, this[index])\n  }\n  return
accumulator\n}\n\n/**\n * Accumulates value starting with the first element and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation]
function that takes the index of an element, current accumulator value\n * and the element itself and calculates the
next accumulator value.\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.reduceIndexed(operation: (index: Int, acc: UShort, UShort) -> UShort): UShort {\n  if (isEmpty())\n
  throw UnsupportedOperationException("Empty array can't be reduced.")\n  var accumulator = this[0]\n  for
(index in 1..lastIndex) {\n    accumulator = operation(index, accumulator, this[index])\n  }\n  return
accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.reduceRight(operation: (UInt, acc: UInt) -> UInt): UInt {\n  var index = lastIndex\n  if (index < 0)
  throw UnsupportedOperationException("Empty array can't be reduced.")\n  var accumulator = get(index--)\n
  while (index >= 0) {\n    accumulator = operation(get(index--), accumulator)\n  }\n  return
accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.reduceRight(operation: (ULong, acc: ULong) -> ULong): ULong {\n  var index = lastIndex\n  if
(index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n  var accumulator =
get(index--)\n  while (index >= 0) {\n    accumulator = operation(get(index--), accumulator)\n  }\n  return
accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.reduceRight(operation: (UByte, acc: UByte) -> UByte): UByte {\n  var index = lastIndex\n  if (index
< 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n  var accumulator = get(index--)\n
  while (index >= 0) {\n    accumulator = operation(get(index--), accumulator)\n  }\n  return
accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.reduceRight(operation: (UShort, acc: UShort) -> UShort): UShort {\n  var index = lastIndex\n  if
(index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n  var accumulator =
get(index--)\n  while (index >= 0) {\n    accumulator = operation(get(index--), accumulator)\n  }\n  return
accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from right to left\n
* to each element with its index in the original array and current accumulator value.\n * @param [operation]
function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the
next accumulator value.\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.reduceRightIndexed(operation: (index: Int, UInt, acc: UInt) -> UInt): UInt {\n  var index = lastIndex\n
  if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n  var accumulator =
get(index--)\n  while (index >= 0) {\n    accumulator = operation(index, get(index), accumulator)\n    --index\n
  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with last element and applying [operation] from
right to left\n * to each element with its index in the original array and current accumulator value.\n * @param
[operation] function that takes the index of an element, the element itself\n * and current accumulator value, and
calculates the next accumulator value.\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.reduceRightIndexed(operation: (index: Int, ULong, acc: ULong) -> ULong): ULong {\n  var index =
lastIndex\n  if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n  var
accumulator = get(index--)\n  while (index >= 0) {\n    accumulator = operation(index, get(index),
accumulator)\n    --index\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with last element
and applying [operation] from right to left\n * to each element with its index in the original array and current
accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n * and
current accumulator value, and calculates the next accumulator value.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.reduceRightIndexed(operation: (index: Int, UByte, acc: UByte) -> UByte): UByte {\n  var index =
lastIndex\n  if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n  var
accumulator = get(index--)\n  while (index >= 0) {\n    accumulator = operation(index, get(index),
accumulator)\n    --index\n  }\n  return accumulator\n}\n\n/**\n * Accumulates value starting with last element
and applying [operation] from right to left\n * to each element with its index in the original array and current
accumulator value.\n * @param [operation] function that takes the index of an element, the element itself\n * and
current accumulator value, and calculates the next accumulator value.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.reduceRightIndexed(operation: (index: Int, UShort, acc: UShort) -> UShort): UShort {\n  var index =
lastIndex\n  if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n  var
accumulator = get(index--)\n  while (index >= 0) {\n    accumulator = operation(index, get(index),
accumulator)\n    --index\n  }\n  return accumulator\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.sumBy(selector: (UInt) -> UInt): UInt {\n  var sum: UInt = 0u\n  for (element in this) {\n    sum +=
selector(element)\n  }\n  return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector] function
applied to each element in the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.sumBy(selector: (ULong) -> UInt): UInt {\n  var sum: UInt = 0u\n  for (element in this) {\n    sum
+= selector(element)\n  }\n  return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector]
function applied to each element in the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.sumBy(selector: (UByte) -> UInt): UInt {\n  var sum: UInt = 0u\n  for (element in this) {\n    sum
+= selector(element)\n  }\n  return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector]
function applied to each element in the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.sumBy(selector: (UShort) -> UInt): UInt {\n  var sum: UInt = 0u\n  for (element in this) {\n
sum += selector(element)\n  }\n  return sum\n}\n\n/**\n * Returns the sum of all values produced by [selector]
function applied to each element in the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.sumByDouble(selector: (UInt) -> Double): Double {\n  var sum: Double = 0.0\n  for (element in this)
{\n    sum += selector(element)\n  }\n  return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.sumByDouble(selector: (ULong) -> Double): Double {\n  var sum: Double = 0.0\n  for (element in
this) {\n    sum += selector(element)\n  }\n  return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.sumByDouble(selector: (UByte) -> Double): Double {\n  var sum: Double = 0.0\n  for (element in

```

```

this) {\n    sum += selector(element)\n  }\n  return sum\n}\n\n/**\n * Returns the sum of all values produced by
[selector] function applied to each element in the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.sumByDouble(selector: (UShort) -> Double): Double {\n  var sum: Double = 0.0\n  for (element in
this) {\n    sum += selector(element)\n  }\n  return sum\n}\n\n/**\n * Returns a list of pairs built from the
elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest
collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UIntArray.zip(other: Array<out
R>): List<Pair<UInt, R>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from
the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest
collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun <R> ULongArray.zip(other:
Array<out R>): List<Pair<ULong, R>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of
pairs built from the elements of `this` array and the [other] array with the same index.\n * The returned list has
length of the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UByteArray.zip(other: Array<out
R>): List<Pair<UByte, R>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built
from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the
shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UShortArray.zip(other:
Array<out R>): List<Pair<UShort, R>> {\n  return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of
values built from the elements of `this` array and the [other] array with the same index\n * using the provided
[transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n * \n
*\n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V>
UIntArray.zip(other: Array<out R>, transform: (a: UInt, b: R) -> V): List<V> {\n  val size = minOf(size,
other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n
}\n  return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V>
ULongArray.zip(other: Array<out R>, transform: (a: ULong, b: R) -> V): List<V> {\n  val size = minOf(size,
other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n
}\n  return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V>
UByteArray.zip(other: Array<out R>, transform: (a: UByte, b: R) -> V): List<V> {\n  val size = minOf(size,
other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n
}\n  return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V>
UShortArray.zip(other: Array<out R>, transform: (a: UShort, b: R) -> V): List<V> {\n  val size = minOf(size,
other.size)\n  val list = ArrayList<V>(size)\n  for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n
}\n  return list\n}\n

```

```

}\n    return list}\n\n/**\n * Returns a list of pairs built from the elements of `this` collection and [other] array with
the same index.\n * The returned list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UIntArray.zip(other:
Iterable<R>): List<Pair<UInt, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs
built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of
the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun <R> ULongArray.zip(other:
Iterable<R>): List<Pair<ULong, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs
built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of
the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UByteArray.zip(other:
Iterable<R>): List<Pair<UByte, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs
built from the elements of `this` collection and [other] array with the same index.\n * The returned list has length of
the shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun <R> UShortArray.zip(other:
Iterable<R>): List<Pair<UShort, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of
values built from the elements of `this` array and the [other] collection with the same index\n * using the provided
[transform] function applied to each pair of elements.\n * The returned list has length of the shortest collection.\n *
\n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V>
UIntArray.zip(other: Iterable<R>, transform: (a: UInt, b: R) -> V): List<V> {\n    val arraySize = size\n    val list =
ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n    var i = 0\n    for (element in other) {\n
if (i >= arraySize) break\n        list.add(transform(this[i++], element))\n    }\n    return list}\n\n/**\n * Returns a
list of values built from the elements of `this` array and the [other] collection with the same index\n * using the
provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest
collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V>
ULongArray.zip(other: Iterable<R>, transform: (a: ULong, b: R) -> V): List<V> {\n    val arraySize = size\n    val
list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n    var i = 0\n    for (element in other)
{\n        if (i >= arraySize) break\n        list.add(transform(this[i++], element))\n    }\n    return list}\n\n/**\n *
Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n *
using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the
shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V>
UByteArray.zip(other: Iterable<R>, transform: (a: UByte, b: R) -> V): List<V> {\n    val arraySize = size\n    val list
= ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n    var i = 0\n    for (element in other) {\n
if (i >= arraySize) break\n        list.add(transform(this[i++], element))\n    }\n    return list}\n\n/**\n * Returns a
list of values built from the elements of `this` array and the [other] collection with the same index\n * using the
provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest
collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V>
UShortArray.zip(other: Iterable<R>, transform: (a: UShort, b: R) -> V): List<V> {\n    val arraySize = size\n    val
list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n    var i = 0\n    for (element in other)
{\n        if (i >= arraySize) break\n        list.add(transform(this[i++], element))\n    }\n    return list}\n\n/**\n *
Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * The
returned list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterable\n

```

```

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun UIntArray.zip(other: UIntArray):
List<Pair<UInt, UInt>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built from
the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the shortest
collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun ULongArray.zip(other: ULongArray):
List<Pair<ULong, ULong>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built
from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the
shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun UByteArray.zip(other: UByteArray):
List<Pair<UByte, UByte>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of pairs built
from the elements of `this` array and the [other] array with the same index.\n * The returned list has length of the
shortest collection.\n * \n * @sample samples.collections.Iterables.Operations.zipIterable\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic infix fun UShortArray.zip(other: UShortArray):
List<Pair<UShort, UShort>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n\n/**\n * Returns a list of values built
from the elements of `this` array and the [other] array with the same index\n * using the provided [transform]
function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V>
UIntArray.zip(other: UIntArray, transform: (a: UInt, b: UInt) -> V): List<V> {\n    val size = minOf(size,
other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n
}\n    return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest array.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V>
ULongArray.zip(other: ULongArray, transform: (a: ULong, b: ULong) -> V): List<V> {\n    val size = minOf(size,
other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n
}\n    return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest array.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V>
UByteArray.zip(other: UByteArray, transform: (a: UByte, b: UByte) -> V): List<V> {\n    val size = minOf(size,
other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n
}\n    return list\n}\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest array.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V>
UShortArray.zip(other: UShortArray, transform: (a: UShort, b: UShort) -> V): List<V> {\n    val size = minOf(size,
other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n
}\n    return list\n}\n\n/**\n * Returns the sum of all elements in the array.\n
*\n@kotlin.jvm.JvmName("sumOfUInt")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Array<out UInt>.sum(): UInt {\n    var sum: UInt = 0u\n    for (element in this) {\n        sum += element\n    }\n
return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n
*\n@kotlin.jvm.JvmName("sumOfULong")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Array<out ULong>.sum(): ULong {\n    var sum: ULong = 0uL\n    for (element in this) {\n        sum += element\n
}\n    return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n

```

```

*\n@kotlin.jvm.JvmName("\sumOfUByte")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Array<out UByte>.sum(): UInt {\n  var sum: UInt = 0u\n  for (element in this) {\n    sum += element\n  }\n  return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n
*\n@kotlin.jvm.JvmName("\sumOfUShort")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Array<out UShort>.sum(): UInt {\n  var sum: UInt = 0u\n  for (element in this) {\n    sum += element\n  }\n  return sum\n}\n\n/**\n * Returns the sum of all elements in the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.sum(): UInt {\n  return storage.sum().toUInt()\n}\n\n/**\n * Returns the sum of all elements in the
array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.sum(): ULong {\n  return storage.sum().toULong()\n}\n\n/**\n * Returns the sum of all elements in
the array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline
fun UByteArray.sum(): UInt {\n  return sumBy { it.toUInt() }\n}\n\n/**\n * Returns the sum of all elements in the
array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.sum(): UInt {\n  return sumBy { it.toUInt() }\n}\n\n",/>\n * Copyright 2010-2019 JetBrains s.r.o.
and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license
that can be found in the license/LICENSE.txt file.\n
*\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("UCollectionsKt")\n\npackage
kotlin.collections\n\n/\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n\nimport kotlin.random.*\nimport
kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns an array of UByte containing all of the
elements of this collection.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Collection<UByte>.toUByteArray(): UByteArray {\n  val result = UByteArray(size)\n  var index = 0\n  for
(element in this)\n    result[index++] = element\n  return result\n}\n\n/**\n * Returns an array of UInt containing
all of the elements of this collection.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Collection<UInt>.toUIntArray(): UIntArray {\n  val result = UIntArray(size)\n  var index = 0\n  for (element in
this)\n    result[index++] = element\n  return result\n}\n\n/**\n * Returns an array of ULong containing all of the
elements of this collection.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Collection<ULong>.toULongArray(): ULongArray {\n  val result = ULongArray(size)\n  var index = 0\n  for
(element in this)\n    result[index++] = element\n  return result\n}\n\n/**\n * Returns an array of UShort
containing all of the elements of this collection.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun Collection<UShort>.toUShortArray():
UShortArray {\n  val result = UShortArray(size)\n  var index = 0\n  for (element in this)\n    result[index++] =
element\n  return result\n}\n\n/**\n * Returns the sum of all elements in the collection.\n
*\n@kotlin.jvm.JvmName("\sumOfUInt")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Iterable<UInt>.sum(): UInt {\n  var sum: UInt = 0u\n  for (element in this) {\n    sum += element\n  }\n  return
sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n
*\n@kotlin.jvm.JvmName("\sumOfULong")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Iterable<ULong>.sum(): ULong {\n  var sum: ULong = 0uL\n  for (element in this) {\n    sum += element\n  }\n  return
sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n
*\n@kotlin.jvm.JvmName("\sumOfUByte")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Iterable<UByte>.sum(): UInt {\n  var sum: UInt = 0u\n  for (element in this) {\n    sum += element\n  }\n  return
sum\n}\n\n/**\n * Returns the sum of all elements in the collection.\n
*\n@kotlin.jvm.JvmName("\sumOfUShort")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Iterable<UShort>.sum(): UInt {\n  var sum: UInt = 0u\n  for (element in this) {\n    sum += element\n  }\n  return
sum\n}\n\n",/>\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("UComparisonsKt")\n\npackage
kotlin.comparisons\n\n/\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:

```



```

https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
import kotlin.random.*
 * Returns the
greater of two values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun maxOf(a: UInt, b:
UInt): UInt {
    return if (a >= b) a else b
}
 * Returns the greater of two values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun maxOf(a: ULong, b: ULong): ULong {
    return if (a >= b) a else b
}
 * Returns the greater of two values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun maxOf(a: UByte, b: UByte): UByte {
    return if (a >= b) a else b
}
 * Returns the greater of two values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun maxOf(a: UShort, b: UShort): UShort {
    return if (a >= b) a else b
}
 * Returns the greater of three values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
maxOf(a: UInt, b: UInt, c: UInt): UInt {
    return maxOf(a, maxOf(b, c))
}
 * Returns the greater of three
values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline
fun maxOf(a: ULong, b: ULong, c: ULong): ULong {
    return maxOf(a, maxOf(b, c))
}
 * Returns the
greater of three values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
maxOf(a: UByte, b: UByte, c: UByte): UByte {
    return maxOf(a, maxOf(b, c))
}
 * Returns the greater
of three values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public
inline fun maxOf(a: UShort, b: UShort, c: UShort): UShort {
    return maxOf(a, maxOf(b, c))
}
 *
Returns the smaller of two values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun
minOf(a: UInt, b: UInt): UInt {
    return if (a <= b) a else b
}
 * Returns the smaller of two values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun minOf(a: ULong, b: ULong): ULong {
    return if (a <= b) a else b
}
 * Returns the smaller of two values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun minOf(a: UByte, b: UByte): UByte {
    return if (a <= b) a else b
}
 * Returns the smaller of two values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun minOf(a: UShort, b: UShort): UShort {
    return if (a <= b) a else b
}
 * Returns the smaller of three values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
minOf(a: UInt, b: UInt, c: UInt): UInt {
    return minOf(a, minOf(b, c))
}
 * Returns the smaller of three
values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline
fun minOf(a: ULong, b: ULong, c: ULong): ULong {
    return minOf(a, minOf(b, c))
}
 * Returns the
smaller of three values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
minOf(a: UByte, b: UByte, c: UByte): UByte {
    return minOf(a, minOf(b, c))
}
 * Returns the smaller
of three values.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public
inline fun minOf(a: UShort, b: UShort, c: UShort): UShort {
    return minOf(a, minOf(b, c))
}
 *
Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is
governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

@file:kotlin.jvm.JvmMultifileClass@file:kotlin.jvm.JvmName("URangesKt")
package
kotlin.ranges
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
import kotlin.random.*
 * Returns a
random element from this range.
 * @throws IllegalArgumentException if this range is empty.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
UIntRange.random(): UInt {
    return random(Random)
}
 * Returns a random element from this range.
 * @throws IllegalArgumentException if this range is empty.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun
ULongRange.random(): ULong {
    return random(Random)
}
 * Returns a random element from this
range using the specified source of randomness.
 * @throws IllegalArgumentException if this range is
empty.
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntRange.random(random:

```

```

Random): UInt {n try {n return random.nextUInt(this)n } catch(e: IllegalArgumentException) {n
throw NoSuchElementException(e.message)n }n}n/n/**n * Returns a random element from this range using
the specified source of randomness.n * n * @throws IllegalArgumentException if this range is empty.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic fun ULongRange.random(random: Random):
ULong {n try {n return random.nextULong(this)n } catch(e: IllegalArgumentException) {n throw
NoSuchElementException(e.message)n }n}n/n/**n * Returns `true` if this range contains the specified
[element].n * n * Always returns `false` if the [element] is `null`.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesn@kotlin.internal.InlineOnlynpublic inline operator fun
UIntRange.contains(element: UInt?): Boolean {n return element != null && contains(element)n}n/n/**n *
Returns `true` if this range contains the specified [element].n * n * Always returns `false` if the [element] is
`null`.n *n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesn@kotlin.internal.InlineOnlynpublic inline
operator fun ULongRange.contains(element: ULong?): Boolean {n return element != null &&
contains(element)n}n/n/**n * Checks if the specified [value] belongs to this range.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic operator fun UIntRange.contains(value:
UByte): Boolean {n return contains(value.toUInt())n}n/n/**n * Checks if the specified [value] belongs to this
range.n *n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic operator fun
ULongRange.contains(value: UByte): Boolean {n return contains(value.toULong())n}n/n/**n * Checks if the
specified [value] belongs to this range.n *n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic
operator fun ULongRange.contains(value: UInt): Boolean {n return contains(value.toULong())n}n/n/**n *
Checks if the specified [value] belongs to this range.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic operator fun UIntRange.contains(value:
ULong): Boolean {n return (value shr UInt.SIZE_BITS) == 0uL && contains(value.toUInt())n}n/n/**n *
Checks if the specified [value] belongs to this range.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic operator fun UIntRange.contains(value:
UShort): Boolean {n return contains(value.toUInt())n}n/n/**n * Checks if the specified [value] belongs to this
range.n *n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic operator fun
ULongRange.contains(value: UShort): Boolean {n return contains(value.toULong())n}n/n/**n * Returns a
progression from this value down to the specified [to] value with the step -1.n * n * The [to] value should be less
than or equal to `this` value.n * If the [to] value is greater than `this` value the returned progression is empty.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic infix fun UByte.downTo(to: UByte):
UIntProgression {n return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)n}n/n/**n * Returns
a progression from this value down to the specified [to] value with the step -1.n * n * The [to] value should be less
than or equal to `this` value.n * If the [to] value is greater than `this` value the returned progression is empty.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic infix fun UInt.downTo(to: UInt):
UIntProgression {n return UIntProgression.fromClosedRange(this, to, -1)n}n/n/**n * Returns a progression
from this value down to the specified [to] value with the step -1.n * n * The [to] value should be less than or equal
to `this` value.n * If the [to] value is greater than `this` value the returned progression is empty.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic infix fun ULong.downTo(to: ULong):
ULongProgression {n return ULongProgression.fromClosedRange(this, to, -1L)n}n/n/**n * Returns a
progression from this value down to the specified [to] value with the step -1.n * n * The [to] value should be less
than or equal to `this` value.n * If the [to] value is greater than `this` value the returned progression is empty.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic infix fun UShort.downTo(to: UShort):
UIntProgression {n return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)n}n/n/**n * Returns
a progression that goes over the same range in the opposite direction with the same step.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic fun UIntProgression.reversed():
UIntProgression {n return UIntProgression.fromClosedRange(last, first, -step)n}n/n/**n * Returns a
progression that goes over the same range in the opposite direction with the same step.n
*/n@SinceKotlin("1.3")n@ExperimentalUnsignedTypesnpublic fun ULongProgression.reversed():

```

```

ULongProgression {
    return ULongProgression.fromClosedRange(last, first, -step)
}

* Returns a progression that goes over the same range with the given step.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UIntProgression.step(step: Int):
    UIntProgression {
    checkStepIsPositive(step > 0, step)
    return UIntProgression.fromClosedRange(first, last, if
        (this.step > 0) step else -step)
}

* Returns a progression that goes over the same range with the given
step.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun ULongProgression.step(step:
    Long): ULongProgression {
    checkStepIsPositive(step > 0, step)
    return
    ULongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}

* Returns a range from this value up to but excluding the specified [to] value.
* If the [to] value is less than or equal to `this` value,
then the returned range is empty.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun
    UByte.until(to: UByte): UIntRange {
    if (to <= UByte.MIN_VALUE) return UIntRange.EMPTY
    return
    this.toUInt() .. (to - 1u).toUInt()
}

* Returns a range from this value up to but excluding the specified [to]
value.
* If the [to] value is less than or equal to `this` value, then the returned range is empty.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun
    UInt.until(to: UInt): UIntRange {
    if (to <= UInt.MIN_VALUE) return UIntRange.EMPTY
    return this .. (to - 1u).toUInt()
}

* Returns a range from this value up to but excluding the specified [to] value.
* If the [to] value is less than or equal to `this` value, then the returned
range is empty.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun
    ULong.until(to: ULong): ULongRange {
    if (to <= ULong.MIN_VALUE) return
    ULongRange.EMPTY
    return this .. (to - 1u).toULong()
}

* Returns a range from this value up to but
excluding the specified [to] value.
* If the [to] value is less than or equal to `this` value, then the returned
range is empty.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun
    UShort.until(to:
    UShort): UIntRange {
    if (to <= UShort.MIN_VALUE) return UIntRange.EMPTY
    return this.toUInt() .. (to -
    1u).toUInt()
}

* Ensures that this value is not less than the specified [minimumValue].
* @return
this value if it's greater than or equal to the [minimumValue] or the
[minimumValue] otherwise.
* @sample
samples.comparisons.ComparableOps.coerceAtLeastUnsigned

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
    UInt.coerceAtLeast(minimumValue: UInt):
    UInt {
    return if (this < minimumValue) minimumValue else this
}

* Ensures that this value is not less
than the specified [minimumValue].
* @return
this value if it's greater than or equal to the [minimumValue]
or the [minimumValue] otherwise.
* @sample
samples.comparisons.ComparableOps.coerceAtLeastUnsigned

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
    ULong.coerceAtLeast(minimumValue:
    ULong): ULong {
    return if (this < minimumValue) minimumValue else this
}

* Ensures that this
value is not less than the specified [minimumValue].
* @return
this value if it's greater than or equal to the
[minimumValue] or the [minimumValue] otherwise.
* @sample
samples.comparisons.ComparableOps.coerceAtLeastUnsigned

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
    UByte.coerceAtLeast(minimumValue:
    UByte): UByte {
    return if (this < minimumValue) minimumValue else this
}

* Ensures that this value
is not less than the specified [minimumValue].
* @return
this value if it's greater than or equal to the
[minimumValue] or the [minimumValue] otherwise.
* @sample
samples.comparisons.ComparableOps.coerceAtLeastUnsigned

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
    UShort.coerceAtLeast(minimumValue:
    UShort): UShort {
    return if (this < minimumValue) minimumValue else this
}

* Ensures that this
value is not greater than the specified [maximumValue].
* @return
this value if it's less than or equal to the
[maximumValue] or the [maximumValue] otherwise.
* @sample
samples.comparisons.ComparableOps.coerceAtMostUnsigned

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
    UInt.coerceAtMost(maximumValue: UInt):
    UInt {
    return if (this > maximumValue) maximumValue else this
}

* Ensures that this value is not
greater than the specified [maximumValue].
* @return
this value if it's less than or equal to the

```

```

[maximumValue] or the [maximumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtMostUnsigned\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULong.coerceAtMost(maximumValue:
ULong): ULong {\n    return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this
value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the
[maximumValue] or the [maximumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtMostUnsigned\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByte.coerceAtMost(maximumValue:
UByte): UByte {\n    return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this
value is not greater than the specified [maximumValue].\n * \n * @return this value if it's less than or equal to the
[maximumValue] or the [maximumValue] otherwise.\n * \n * @sample
samples.comparisons.ComparableOps.coerceAtMostUnsigned\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShort.coerceAtMost(maximumValue:
UShort): UShort {\n    return if (this > maximumValue) maximumValue else this\n}\n\n/**\n * Ensures that this
value lies in the specified range [minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range,
or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than
[maximumValue].\n * \n * @sample samples.comparisons.ComparableOps.coerceInUnsigned\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UInt.coerceIn(minimumValue: UInt,
maximumValue: UInt): UInt {\n    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot
coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")\n    if (this <
minimumValue) return minimumValue\n    if (this > maximumValue) return maximumValue\n    return
this\n}\n\n/**\n * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].\n * \n *
@return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or
[maximumValue] if this value is greater than [maximumValue].\n * \n * @sample
samples.comparisons.ComparableOps.coerceInUnsigned\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULong.coerceIn(minimumValue: ULong,
maximumValue: ULong): ULong {\n    if (minimumValue > maximumValue) throw
IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")\n    if (this < minimumValue) return minimumValue\n    if (this > maximumValue)
return maximumValue\n    return this\n}\n\n/**\n * Ensures that this value lies in the specified range
[minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value
is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample
samples.comparisons.ComparableOps.coerceInUnsigned\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByte.coerceIn(minimumValue: UByte,
maximumValue: UByte): UByte {\n    if (minimumValue > maximumValue) throw
IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")\n    if (this < minimumValue) return minimumValue\n    if (this > maximumValue)
return maximumValue\n    return this\n}\n\n/**\n * Ensures that this value lies in the specified range
[minimumValue]..[maximumValue].\n * \n * @return this value if it's in the range, or [minimumValue] if this value
is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].\n * \n * @sample
samples.comparisons.ComparableOps.coerceInUnsigned\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShort.coerceIn(minimumValue: UShort,
maximumValue: UShort): UShort {\n    if (minimumValue > maximumValue) throw
IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")\n    if (this < minimumValue) return minimumValue\n    if (this > maximumValue)
return maximumValue\n    return this\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * \n *
@return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive`
if this value is greater than `range.endInclusive`.\n * \n * @sample

```

```

samples.comparisons.ComparableOps.coerceInUnsigned\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UInt.coerceIn(range: ClosedRange<UInt>):
UInt {\n    if (range is ClosedFloatingPointRange) {\n        return this.coerceIn<UInt>(range)\n    }\n    if
(range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")\n    return
when {\n        this < range.start -> range.start\n        this > range.endInclusive -> range.endInclusive\n        else ->
this\n    }\n}\n\n/**\n * Ensures that this value lies in the specified [range].\n * @return this value if it's in the
[range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than
`range.endInclusive`.\n */\n * @sample samples.comparisons.ComparableOps.coerceInUnsigned\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULong.coerceIn(range:
ClosedRange<ULong>): ULong {\n    if (range is ClosedFloatingPointRange) {\n        return
this.coerceIn<ULong>(range)\n    }\n    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value
to an empty range: $range.")\n    return when {\n        this < range.start -> range.start\n        this >
range.endInclusive -> range.endInclusive\n        else -> this\n    }\n}\n\n"/**\n * Copyright 2010-2019 JetBrains
s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0
license that can be found in the license/LICENSE.txt file.\n
*\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("USequencesKt")\n\npackage
kotlin.sequences\n\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n\nimport kotlin.random.*\n\n/**\n * Returns the
sum of all elements in the sequence.\n * The operation is _terminal_.\n
*\n@kotlin.jvm.JvmName("sumOfUInt")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Sequence<UInt>.sum(): UInt {\n    var sum: UInt = 0u\n    for (element in this) {\n        sum += element\n    }\n
return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n * The operation is _terminal_.\n
*\n@kotlin.jvm.JvmName("sumOfULong")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Sequence<ULong>.sum(): ULong {\n    var sum: ULong = 0uL\n    for (element in this) {\n        sum += element\n
    }\n    return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n * The operation is
_terminal_.\n
*\n@kotlin.jvm.JvmName("sumOfUByte")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Sequence<UByte>.sum(): UInt {\n    var sum: UInt = 0u\n    for (element in this) {\n        sum += element\n    }\n
return sum\n}\n\n/**\n * Returns the sum of all elements in the sequence.\n * The operation is _terminal_.\n
*\n@kotlin.jvm.JvmName("sumOfUShort")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun
Sequence<UShort>.sum(): UInt {\n    var sum: UInt = 0u\n    for (element in this) {\n        sum += element\n    }\n
return sum\n}\n\n"/**\r\n * Copyright 2010-2015 JetBrains s.r.o.\r\n */\r\n * Licensed under the Apache License,
Version 2.0 (the "License");\r\n * you may not use this file except in compliance with the License.\r\n * You may
obtain a copy of the License at\r\n * http://www.apache.org/licenses/LICENSE-2.0\r\n * Unless required
by applicable law or agreed to in writing, software\r\n * distributed under the License is distributed on an "AS IS"
BASIS,\r\n * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.\r\n * See
the License for the specific language governing permissions and\r\n * limitations under the License.\r\n
*\n@r\n\npackage kotlin.annotation\n\nimport kotlin.annotation.AnnotationTarget.*\n\n/**\r\n\r\n/**\r\n * Contains the
list of code elements which are the possible annotation targets\n * @r\npublic enum class AnnotationTarget {\r\n
/**\r\n * Class, interface or object, annotation class is also included\n * @r\n CLASS,\r\n /**\r\n * Annotation class only\n * @r\n
ANNOTATION_CLASS,\r\n /**\r\n * Generic type parameter (unsupported yet)\n * @r\n TYPE_PARAMETER,\r\n\r\n
/**\r\n * Property\n * @r\n PROPERTY,\r\n /**\r\n * Field, including property's backing field\n * @r\n FIELD,\r\n /**\r\n * Local
variable\n * @r\n LOCAL_VARIABLE,\r\n /**\r\n * Value parameter of a function or a constructor\n * @r\n
VALUE_PARAMETER,\r\n /**\r\n * Constructor only (primary or secondary)\n * @r\n CONSTRUCTOR,\r\n /**\r\n *
Function (constructors are not included)\n * @r\n FUNCTION,\r\n /**\r\n * Property getter only\n * @r\n
PROPERTY_GETTER,\r\n /**\r\n * Property setter only\n * @r\n PROPERTY_SETTER,\r\n /**\r\n * Type usage\n * @r\n
TYPE,\r\n /**\r\n * Any expression\n * @r\n EXPRESSION,\r\n /**\r\n * File\n * @r\n FILE,\r\n /**\r\n * Type alias\n * @r\n
TYPEALIAS\r\n}\r\n\r\n/**\r\n * Contains the list of possible annotation's

```

```

retentions.\r\n * \r\n * Determines how an annotation is stored in binary output.\r\n * \r\n\r\npublic enum class
AnnotationRetention {\r\n    /** Annotation isn't stored in binary output */\r\n    SOURCE,\r\n    /** Annotation is
stored in binary output, but invisible for reflection */\r\n    BINARY,\r\n    /** Annotation is stored in binary output
and visible for reflection (default retention) */\r\n    RUNTIME\r\n}\r\n\r\n/**\r\n * This meta-annotation indicates
the kinds of code elements which are possible targets of an annotation.\r\n * \r\n * If the target meta-annotation is not
present on an annotation declaration, the annotation is applicable to the following elements:\r\n * [CLASS],
[PROPERTY], [FIELD], [LOCAL_VARIABLE], [VALUE_PARAMETER], [CONSTRUCTOR], [FUNCTION],
[PROPERTY_GETTER], [PROPERTY_SETTER].\r\n * \r\n * @property allowedTargets list of allowed annotation
targets\r\n * \r\n * @Target(AnnotationTarget.ANNOTATION_CLASS)\r\n * \r\n * @MustBeDocumented\r\n\r\npublic
annotation class Target(vararg val allowedTargets: AnnotationTarget)\r\n\r\n/**\r\n * This meta-annotation
determines whether an annotation is stored in binary output and visible for reflection. By default, both are true.\r\n
\r\n * \r\n * @property value necessary annotation retention (RUNTIME, BINARY or SOURCE)\r\n
\r\n * \r\n * @Target(AnnotationTarget.ANNOTATION_CLASS)\r\n\r\npublic annotation class Retention(val value:
AnnotationRetention = AnnotationRetention.RUNTIME)\r\n\r\n/**\r\n * This meta-annotation determines that an
annotation is applicable twice or more on a single code element\r\n
\r\n * \r\n * @Target(AnnotationTarget.ANNOTATION_CLASS)\r\n\r\npublic annotation class Repeatable\r\n\r\n\r\n/**\r\n *
This meta-annotation determines that an annotation is a part of public API and therefore should be included in the
generated\r\n * documentation for the element to which the annotation is applied.\r\n
\r\n * \r\n * @Target(AnnotationTarget.ANNOTATION_CLASS)\r\n\r\npublic annotation class
MustBeDocumented\r\n", "/*\r\n * Copyright 2010-2015 JetBrains s.r.o.\r\n * \r\n * Licensed under the Apache
License, Version 2.0 (the "License");\r\n * you may not use this file except in compliance with the License.\r\n *
You may obtain a copy of the License at\r\n * \r\n * http://www.apache.org/licenses/LICENSE-2.0\r\n * \r\n * Unless
required by applicable law or agreed to in writing, software\r\n * distributed under the License is distributed on an
"AS IS" BASIS,\r\n * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied.\r\n * See the License for the specific language governing permissions and\r\n * limitations under the
License.\r\n * \r\n * \r\n\r\npackage kotlin.collections\r\n\r\n\r\nimport kotlin.internal.PlatformDependent\r\n\r\n\r\n\r\n/**\r\n * Classes that inherit from this interface can be represented as a sequence of elements that can\r\n * be iterated
over.\r\n * @param T the type of element being iterated over. The iterator is covariant on its element type.\r\n
\r\n * \r\n\r\npublic interface Iterable<out T> {\r\n    /**\r\n     * Returns an iterator over the elements of this object.\r\n
\r\n     * \r\n     * public operator fun iterator(): Iterator<T>\r\n}\r\n\r\n\r\n\r\n/**\r\n * Classes that inherit from this interface can be
represented as a sequence of elements that can\r\n * be iterated over and that supports removing elements during
iteration.\r\n * @param T the type of element being iterated over. The mutable iterator is invariant on its element
type.\r\n * \r\n\r\npublic interface MutableIterable<out T> : Iterable<T> {\r\n    /**\r\n     * Returns an iterator over the
elements of this sequence that supports removing elements during iteration.\r\n     * \r\n     * override fun iterator():
MutableIterator<T>\r\n}\r\n\r\n\r\n\r\n/**\r\n * A generic collection of elements. Methods in this interface support only
read-only access to the collection;\r\n * read/write access is supported through the [MutableCollection] interface.\r\n
\r\n * @param E the type of elements contained in the collection. The collection is covariant on its element type.\r\n
\r\n * \r\n\r\npublic interface Collection<out E> : Iterable<E> {\r\n    // Query Operations\r\n    /**\r\n     * Returns the size
of the collection.\r\n     * \r\n     * public val size: Int\r\n\r\n     * Returns `true` if the collection is empty
(contains no elements), `false` otherwise.\r\n     * \r\n     * public fun isEmpty(): Boolean\r\n\r\n     * Checks if
the specified element is contained in this collection.\r\n     * \r\n     * public operator fun contains(element:
@UnsafeVariance E): Boolean\r\n\r\n     * override fun iterator(): Iterator<E>\r\n\r\n     // Bulk Operations\r\n     /**\r\n     * Checks if all elements in the specified collection are contained in this collection.\r\n     * \r\n     * public fun
containsAll(elements: Collection<@UnsafeVariance E>): Boolean\r\n}\r\n\r\n\r\n\r\n/**\r\n * A generic collection of
elements that supports adding and removing elements.\r\n * \r\n * @param E the type of elements contained in the
collection. The mutable collection is invariant on its element type.\r\n * \r\n\r\npublic interface MutableCollection<E> :
Collection<E>, MutableIterable<E> {\r\n    // Query Operations\r\n     * override fun iterator():
MutableIterator<E>\r\n\r\n     // Modification Operations\r\n     /**\r\n     * Adds the specified element to the

```

```

collection.\r\n    * \r\n    * @return `true` if the element has been added, `false` if the collection does not support
duplicates\r\n    * and the element is already contained in the collection.\r\n    * \r\n    public fun add(element: E):
Boolean\r\n\r\n    /**\r\n    * Removes a single instance of the specified element from this\r\n    * collection, if it is
present.\r\n    * \r\n    * @return `true` if the element has been successfully removed; `false` if it was not present in
the collection.\r\n    * \r\n    public fun remove(element: E): Boolean\r\n\r\n    // Bulk Modification Operations\r\n    /**\r\n    * Adds all of the elements of the specified collection to this collection.\r\n    * \r\n    * @return `true` if
any of the specified elements was added to the collection, `false` if the collection was not modified.\r\n    * \r\n
public fun addAll(elements: Collection<E>): Boolean\r\n\r\n    /**\r\n    * Removes all of this collection's elements
that are also contained in the specified collection.\r\n    * \r\n    * @return `true` if any of the specified elements was
removed from the collection, `false` if the collection was not modified.\r\n    * \r\n    public fun
removeAll(elements: Collection<E>): Boolean\r\n\r\n    /**\r\n    * Retains only the elements in this collection that
are contained in the specified collection.\r\n    * \r\n    * @return `true` if any element was removed from the
collection, `false` if the collection was not modified.\r\n    * \r\n    public fun retainAll(elements: Collection<E>):
Boolean\r\n\r\n    /**\r\n    * Removes all elements from this collection.\r\n    * \r\n    public fun clear():
Unit\r\n\r\n    /**\r\n    * A generic ordered collection of elements. Methods in this interface support only read-only
access to the list;\r\n    * read/write access is supported through the [MutableList] interface.\r\n    * @param E the type
of elements contained in the list. The list is covariant on its element type.\r\n    * \r\n    public interface List<out E> :
Collection<E> {\r\n    // Query Operations\r\n\r\n    override val size: Int\r\n    override fun isEmpty(): Boolean\r\n
override fun contains(element: @UnsafeVariance E): Boolean\r\n    override fun iterator(): Iterator<E>\r\n\r\n    //
Bulk Operations\r\n    override fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean\r\n\r\n    //
Positional Access Operations\r\n    /**\r\n    * Returns the element at the specified index in the list.\r\n    * \r\n
public operator fun get(index: Int): E\r\n\r\n    // Search Operations\r\n    /**\r\n    * Returns the index of the first
occurrence of the specified element in the list, or -1 if the specified\r\n    * element is not contained in the list.\r\n
\r\n    * \r\n    public fun indexOf(element: @UnsafeVariance E): Int\r\n\r\n    /**\r\n    * Returns the index of the last
occurrence of the specified element in the list, or -1 if the specified\r\n    * element is not contained in the list.\r\n
\r\n    * \r\n    public fun lastIndexOf(element: @UnsafeVariance E): Int\r\n\r\n    // List Iterators\r\n    /**\r\n    * Returns
a list iterator over the elements in this list (in proper sequence).\r\n    * \r\n    public fun listIterator():
ListIterator<E>\r\n\r\n    /**\r\n    * Returns a list iterator over the elements in this list (in proper sequence), starting
at the specified [index].\r\n    * \r\n    public fun listIterator(index: Int): ListIterator<E>\r\n\r\n    // View\r\n
\r\n    /**\r\n    * Returns a view of the portion of this list between the specified [fromIndex] (inclusive) and [toIndex]
(exclusive).\r\n    * \r\n    * The returned list is backed by this list, so non-structural changes in the returned list are reflected
in this list, and vice-versa.\r\n    * \r\n    * Structural changes in the base list make the behavior of the view
undefined.\r\n    * \r\n    public fun subList(fromIndex: Int, toIndex: Int): List<E>\r\n\r\n    /**\r\n    * A generic
ordered collection of elements that supports adding and removing elements.\r\n    * @param E the type of elements
contained in the list. The mutable list is invariant on its element type.\r\n    * \r\n    public interface MutableList<E> :
List<E>, MutableCollection<E> {\r\n    // Modification Operations\r\n    /**\r\n    * Adds the specified element to
the end of this list.\r\n    * \r\n    * @return `true` because the list is always modified as the result of this
operation.\r\n    * \r\n    override fun add(element: E): Boolean\r\n\r\n    override fun remove(element: E):
Boolean\r\n\r\n    // Bulk Modification Operations\r\n    /**\r\n    * Adds all of the elements of the specified
collection to the end of this list.\r\n    * \r\n    * The elements are appended in the order they appear in the [elements]
collection.\r\n    * \r\n    * @return `true` if the list was changed as the result of the operation.\r\n    * \r\n
override fun addAll(elements: Collection<E>): Boolean\r\n\r\n    /**\r\n    * Inserts all of the elements of the specified
collection [elements] into this list at the specified [index].\r\n    * \r\n    * @return `true` if the list was changed as
the result of the operation.\r\n    * \r\n    public fun addAll(index: Int, elements: Collection<E>): Boolean\r\n\r\n
override fun removeAll(elements: Collection<E>): Boolean\r\n    override fun retainAll(elements: Collection<E>):
Boolean\r\n    override fun clear(): Unit\r\n\r\n    // Positional Access Operations\r\n    /**\r\n    * Replaces the
element at the specified position in this list with the specified element.\r\n    * \r\n    * @return the element
previously at the specified position.\r\n    * \r\n    public operator fun set(index: Int, element: E): E\r\n\r\n    /**\r\n

```

```

* Inserts an element into the list at the specified [index].\r\n    */\r\n    public fun add(index: Int, element: E):
Unit\r\n\r\n    /**\r\n    * Removes an element at the specified [index] from the list.\r\n    */\r\n    * @return the
element that has been removed.\r\n    */\r\n    public fun removeAt(index: Int): E\r\n\r\n    // List Iterators\r\n
override fun listIterator(): MutableListIterator<E>\r\n\r\n    override fun listIterator(index: Int):
MutableListIterator<E>\r\n\r\n    // View\r\n    override fun subList(fromIndex: Int, toIndex: Int):
MutableList<E>\r\n\r\n}\r\n\r\n/**\r\n * A generic unordered collection of elements that does not support duplicate
elements.\r\n * Methods in this interface support only read-only access to the set;\r\n * read/write access is
supported through the [MutableSet] interface.\r\n * @param E the type of elements contained in the set. The set is
covariant on its element type.\r\n */\r\n\r\npublic interface Set<out E> : Collection<E> {\r\n    // Query
Operations\r\n\r\n    override val size: Int\r\n    override fun isEmpty(): Boolean\r\n    override fun contains(element:
@UnsafeVariance E): Boolean\r\n    override fun iterator(): Iterator<E>\r\n\r\n    // Bulk Operations\r\n    override
fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean\r\n}\r\n\r\n/**\r\n * A generic unordered
collection of elements that does not support duplicate elements, and supports\r\n * adding and removing
elements.\r\n * @param E the type of elements contained in the set. The mutable set is invariant on its element
type.\r\n */\r\n\r\npublic interface MutableSet<E> : Set<E>, MutableCollection<E> {\r\n    // Query Operations\r\n
override fun iterator(): MutableIterator<E>\r\n\r\n    // Modification Operations\r\n\r\n    /**\r\n    * Adds the
specified element to the set.\r\n    */\r\n    * @return `true` if the element has been added, `false` if the element is
already contained in the set.\r\n    */\r\n    override fun add(element: E): Boolean\r\n\r\n    override fun
remove(element: E): Boolean\r\n\r\n    // Bulk Modification Operations\r\n\r\n    override fun addAll(elements:
Collection<E>): Boolean\r\n    override fun removeAll(elements: Collection<E>): Boolean\r\n    override fun
retainAll(elements: Collection<E>): Boolean\r\n    override fun clear(): Unit\r\n\r\n}\r\n\r\n/**\r\n * A collection that
holds pairs of objects (keys and values) and supports efficiently retrieving\r\n * the value corresponding to each key.
Map keys are unique; the map holds only one value for each key.\r\n * Methods in this interface support only read-
only access to the map; read-write access is supported through\r\n * the [MutableMap] interface.\r\n * @param K
the type of map keys. The map is invariant on its key type, as it\r\n *      can accept key as a parameter (of
[containsKey] for example) and return it in [keys] set.\r\n * @param V the type of map values. The map is covariant
on its value type.\r\n */\r\n\r\npublic interface Map<K, out V> {\r\n    // Query Operations\r\n    /**\r\n    * Returns the
number of key/value pairs in the map.\r\n    */\r\n    public val size: Int\r\n\r\n    /**\r\n    * Returns `true` if the map
is empty (contains no elements), `false` otherwise.\r\n    */\r\n    public fun isEmpty(): Boolean\r\n\r\n    /**\r\n    *
Returns `true` if the map contains the specified [key].\r\n    */\r\n    public fun containsKey(key: K): Boolean\r\n\r\n
    /**\r\n    * Returns `true` if the map maps one or more keys to the specified [value].\r\n    */\r\n    public fun
containsValue(value: @UnsafeVariance V): Boolean\r\n\r\n    /**\r\n    * Returns the value corresponding to the
given [key], or `null` if such a key is not present in the map.\r\n    */\r\n    public operator fun get(key: K): V?\r\n\r\n
    /**\r\n    * Returns the value corresponding to the given [key], or [defaultValue] if such a key is not present in the
map.\r\n    */\r\n    * @since JDK 1.8\r\n    */\r\n    @SinceKotlin("1.1")\r\n    @PlatformDependent\r\n    public
fun getOrDefault(key: K, defaultValue: @UnsafeVariance V): V {\r\n        // See default implementation in JDK
sources\r\n        return null as V\r\n    }\r\n\r\n    // Views\r\n    /**\r\n    * Returns a read-only [Set] of all keys in
this map.\r\n    */\r\n    public val keys: Set<K>\r\n\r\n    /**\r\n    * Returns a read-only [Collection] of all values in
this map. Note that this collection may contain duplicate values.\r\n    */\r\n    public val values:
Collection<V>\r\n\r\n    /**\r\n    * Returns a read-only [Set] of all key/value pairs in this map.\r\n    */\r\n    public
val entries: Set<Map.Entry<K, V>>\r\n\r\n    /**\r\n    * Represents a key/value pair held by a [Map].\r\n    */\r\n
    public interface Entry<out K, out V> {\r\n        /**\r\n        * Returns the key of this key/value pair.\r\n        */\r\n
        public val key: K\r\n\r\n        /**\r\n        * Returns the value of this key/value pair.\r\n        */\r\n
        public val value: V\r\n    }\r\n\r\n}\r\n\r\n/**\r\n * A modifiable collection that holds pairs of objects (keys and values) and
supports efficiently retrieving\r\n * the value corresponding to each key. Map keys are unique; the map holds only
one value for each key.\r\n * @param K the type of map keys. The map is invariant on its key type.\r\n * @param V
the type of map values. The mutable map is invariant on its value type.\r\n */\r\n\r\npublic interface MutableMap<K,
V> : Map<K, V> {\r\n    // Modification Operations\r\n    /**\r\n    * Associates the specified [value] with the

```



```

specified [key] in the map.
    * @return the previous value associated with the key, or `null` if the key
was not present in the map.
    * @return public fun put(key: K, value: V): V?
    /** Removes the
specified key and its corresponding value from this map.
    * @return the previous value associated with
the key, or `null` if the key was not present in the map.
    * @return public fun remove(key: K): V?
    /**
    * Removes the entry for the specified key only if it is mapped to the specified value.
    * @return true
if entry was removed
    * @SinceKotlin("1.1")
    * @PlatformDependent
    * public fun remove(key:
K, value: V): Boolean
    {
    // See default implementation in JDK sources
    return true
    }
Bulk Modification Operations
    * Updates this map with key/value pairs from the specified map
[from].
    * public fun putAll(from: Map<out K, V>): Unit
    /** Removes all elements
from this map.
    * public fun clear(): Unit
    // Views
    /** Returns a [MutableSet] of all
keys in this map.
    * override val keys: MutableSet<K>
    /** Returns a
[MutableCollection] of all values in this map. Note that this collection may contain duplicate values.
    * override val values: MutableCollection<V>
    /** Returns a [MutableSet] of all key/value pairs in this
map.
    * override val entries: MutableSet<MutableMap.MutableEntry<K, V>>
    /**
Represents a key/value pair held by a [MutableMap].
    * public interface MutableEntry<K, V> :
Map.Entry<K, V>
    /**
    * Changes the value associated with the key of this entry.
    * @return the previous value corresponding to the key.
    * public fun setValue(newValue: V): V
    }
}
    /** Copyright 2010-2016 JetBrains s.r.o.
    * Licensed under the Apache License, Version
2.0 (the "License");
    * you may not use this file except in compliance with the License.
    * You may obtain a
copy of the License at
    * http://www.apache.org/licenses/LICENSE-2.0
    * Unless required by
applicable law or agreed to in writing, software
    * distributed under the License is distributed on an "AS IS"
BASIS,
    * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
    * See
the License for the specific language governing permissions and
    * limitations under the License.
    * @package kotlin.internal
    /** Specifies that the corresponding type parameter is not used for
unsafe operations such as casts or 'is' checks
    * That means it's completely safe to use generic types as argument
for such parameter.
    * @Target(AnnotationTarget.TYPE_PARAMETER)
    * @Retention(AnnotationRetention.BINARY)
    * internal
annotation class PureReifiable
    /** Specifies that the corresponding built-in method exists depending on
platform.
    * Current implementation for JVM looks whether method with same JVM descriptor exists in the
module JDK.
    * For example MutableMap.remove(K, V) available only if corresponding
method
'java/util/Map.remove(Ljava/lang/Object;Ljava/lang/Object;)Z' is defined in JDK (i.e. for major versions >= 8)
    * @Target(AnnotationTarget.FUNCTION)
    * @Retention(AnnotationRetention.BINARY)
    * internal
annotation class PlatformDependent
    /** Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.
    * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.
    * @package kotlin.internal
    * // a mod b (in arithmetical sense)
    * private fun
mod(a: Int, b: Int): Int
    {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
    }
    * private fun
mod(a: Long, b: Long): Long
    {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
    }
    * // (a -
b) mod c
    * private fun differenceModulo(a: Int, b: Int, c: Int): Int
    {
    return mod(mod(a, c) - mod(b, c),
c)
    }
    * private fun differenceModulo(a: Long, b: Long, c: Long): Long
    {
    return mod(mod(a, c) - mod(b,
c), c)
    }
    /** Calculates the final element of a bounded arithmetic progression, i.e. the last element of
the progression which is in the range
    * from [start] to [end] in case of a positive [step], or from [end] to [start] in
case of a negative
    * [step].
    * No validation on passed parameters is performed. The given parameters
should satisfy the condition:
    * - either `step > 0` and `start <= end`,
    * - or `step < 0` and `start >=
end`.
    * @param start first element of the progression
    * @param end ending bound for the
progression
    * @param step increment, or difference of successive elements in the progression
    * @return the
final element of the progression
    * @suppress
    * @PublishedApi
    * internal fun
getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {
    step > 0 -> if (start >= end) end else end -
differenceModulo(end, start, step)
    step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -

```

```

step)\r\n else -> throw kotlin.IllegalArgumentException("Step is zero.\")\r\n}\r\n\r\n**\r\n * Calculates the final
element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range\r\n * from
[start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative\r\n * [step].\r\n * \r\n * No
validation on passed parameters is performed. The given parameters should satisfy the condition:\r\n * \r\n * - either
`step > 0` and `start <= end`,\r\n * - or `step < 0` and `start >= end`.\r\n * \r\n * @param start first element of the
progression\r\n * @param end ending bound for the progression\r\n * @param step increment, or difference of
successive elements in the progression\r\n * @return the final element of the progression\r\n * @suppress\r\n
*/\r\n@PublishedApi\r\ninternal fun getProgressionLastElement(start: Long, end: Long, step: Long): Long = when
{\r\n step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)\r\n step < 0 -> if (start <= end)
end else end + differenceModulo(start, end, -step)\r\n else -> throw kotlin.IllegalArgumentException("Step is
zero.\")\r\n}\r\n\r\n","/*\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt
file.\r\n * \r\n\r\n// Auto-generated file. DO NOT EDIT!\r\n\r\npackage kotlin.collections\r\n\r\n** An iterator over
a sequence of values of type `Byte`. */\r\npublic abstract class ByteIterator : Iterator<Byte> {\r\n override final fun
next() = nextByte()\r\n\r\n /** Returns the next value in the sequence without boxing. */\r\n public abstract fun
nextByte(): Byte\r\n}\r\n\r\n** An iterator over a sequence of values of type `Char`. */\r\npublic abstract class
CharIterator : Iterator<Char> {\r\n override final fun next() = nextChar()\r\n\r\n /** Returns the next value in the
sequence without boxing. */\r\n public abstract fun nextChar(): Char\r\n}\r\n\r\n** An iterator over a sequence of
values of type `Short`. */\r\npublic abstract class ShortIterator : Iterator<Short> {\r\n override final fun next() =
nextShort()\r\n\r\n /** Returns the next value in the sequence without boxing. */\r\n public abstract fun
nextShort(): Short\r\n}\r\n\r\n** An iterator over a sequence of values of type `Int`. */\r\npublic abstract class
IntIterator : Iterator<Int> {\r\n override final fun next() = nextInt()\r\n\r\n /** Returns the next value in the
sequence without boxing. */\r\n public abstract fun nextInt(): Int\r\n}\r\n\r\n** An iterator over a sequence of
values of type `Long`. */\r\npublic abstract class LongIterator : Iterator<Long> {\r\n override final fun next() =
nextLong()\r\n\r\n /** Returns the next value in the sequence without boxing. */\r\n public abstract fun
nextLong(): Long\r\n}\r\n\r\n** An iterator over a sequence of values of type `Float`. */\r\npublic abstract class
FloatIterator : Iterator<Float> {\r\n override final fun next() = nextFloat()\r\n\r\n /** Returns the next value in
the sequence without boxing. */\r\n public abstract fun nextFloat(): Float\r\n}\r\n\r\n** An iterator over a
sequence of values of type `Double`. */\r\npublic abstract class DoubleIterator : Iterator<Double> {\r\n override
final fun next() = nextDouble()\r\n\r\n /** Returns the next value in the sequence without boxing. */\r\n public
abstract fun nextDouble(): Double\r\n}\r\n\r\n** An iterator over a sequence of values of type `Boolean`.
*/\r\npublic abstract class BooleanIterator : Iterator<Boolean> {\r\n override final fun next() =
nextBoolean()\r\n\r\n /** Returns the next value in the sequence without boxing. */\r\n public abstract fun
nextBoolean(): Boolean\r\n}\r\n\r\n","/*\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming
Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n * \r\n\r\n// Auto-generated file. DO NOT EDIT!\r\n\r\npackage
kotlin.ranges\r\n\r\n**\r\n * An iterator over a progression of values of type `Char`. \r\n * @property step the
number by which the value is incremented on each step.\r\n */\r\ninternal class CharProgressionIterator(first: Char,
last: Char, val step: Int) : CharIterator() {\r\n private val finalElement = last.toInt()\r\n private var hasNext:
Boolean = if (step > 0) first <= last else first >= last\r\n private var next = if (hasNext) first.toInt() else
finalElement\r\n override fun hasNext(): Boolean = hasNext\r\n\r\n override fun nextChar(): Char {\r\n
val value = next\r\n if (value == finalElement) {\r\n if (!hasNext) throw
kotlin.NoSuchElementException()\r\n hasNext = false\r\n }\r\n else {\r\n next += step\r\n
}\r\n return value.toChar()\r\n }\r\n}\r\n\r\n**\r\n * An iterator over a progression of values of type `Int`. \r\n
* @property step the number by which the value is incremented on each step.\r\n */\r\ninternal class
IntProgressionIterator(first: Int, last: Int, val step: Int) : IntIterator() {\r\n private val finalElement = last\r\n
private var hasNext: Boolean = if (step > 0) first <= last else first >= last\r\n private var next = if (hasNext) first
else finalElement\r\n override fun hasNext(): Boolean = hasNext\r\n\r\n override fun nextInt(): Int {\r\n

```

```

val value = next\r\n    if (value == finalElement) {\r\n        if (!hasNext) throw
kotlin.NoSuchElementException()\r\n        hasNext = false\r\n    }\r\n    else {\r\n        next += step\r\n
}\r\n    return value\r\n    }\r\n}\r\n\r\n/**\r\n * An iterator over a progression of values of type `Long`.\r\n *
@property step the number by which the value is incremented on each step.\r\n */\r\ninternal class
LongProgressionIterator(first: Long, last: Long, val step: Long) : LongIterator() {\r\n    private val finalElement =
last\r\n    private var hasNext: Boolean = if (step > 0) first <= last else first >= last\r\n    private var next = if
(hasNext) first else finalElement\r\n\r\n    override fun hasNext(): Boolean = hasNext\r\n\r\n    override fun
nextLong(): Long {\r\n        val value = next\r\n        if (value == finalElement) {\r\n            if (!hasNext) throw
kotlin.NoSuchElementException()\r\n            hasNext = false\r\n        }\r\n        else {\r\n            next += step\r\n
}\r\n        return value\r\n    }\r\n}\r\n\r\n"/**\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming
Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\n// Auto-generated file. DO NOT EDIT!\r\n\r\npackage
kotlin.ranges\r\n\r\nimport kotlin.internal.getProgressionLastElement\r\n\r\n/**\r\n * A progression of values of
type `Char`.\r\n */\r\npublic open class CharProgression\r\n    internal constructor(\r\n        (\r\n            start: Char,\r\n\r\n
endInclusive: Char,\r\n            step: Int\r\n        ) : Iterable<Char> {\r\n    init {\r\n        if (step == 0) throw
kotlin.IllegalArgumentException("Step must be non-zero.")\r\n        if (step == Int.MIN_VALUE) throw
kotlin.IllegalArgumentException("Step must be greater than Int.MIN_VALUE to avoid overflow on
negation.")\r\n    }\r\n\r\n    /**\r\n     * The first element in the progression.\r\n     */\r\n    public val first: Char =
start\r\n\r\n    /**\r\n     * The last element in the progression.\r\n     */\r\n    public val last: Char =
getProgressionLastElement(start.toInt(), endInclusive.toInt(), step).toChar()\r\n\r\n    /**\r\n     * The step of the
progression.\r\n     */\r\n    public val step: Int = step\r\n\r\n    override fun iterator(): CharIterator =
CharProgressionIterator(first, last, step)\r\n\r\n    /** Checks if the progression is empty. */\r\n    public open fun
isEmpty(): Boolean = if (step > 0) first > last else first < last\r\n\r\n    override fun equals(other: Any?): Boolean
= \r\n        other is CharProgression && (isEmpty() && other.isEmpty()) ||\r\n        first == other.first && last ==
other.last && step == other.step)\r\n\r\n    override fun hashCode(): Int = \r\n        if (isEmpty()) -1 else (31 * (31 *
first.toInt() + last.toInt()) + step)\r\n\r\n    override fun toString(): String = if (step > 0) "$first..$last step $step" else
"$first downTo $last step ${-step}"\r\n\r\n    companion object {\r\n        /**\r\n         * Creates CharProgression
within the specified bounds of a closed range.\r\n         * The progression starts with the [rangeStart] value and
goes toward the [rangeEnd] value not excluding it, with the specified [step].\r\n         * In order to go backwards the
[step] must be negative.\r\n         */\r\n        * [step] must be greater than `Int.MIN_VALUE` and not equal to
zero.\r\n         */\r\n        public fun fromClosedRange(rangeStart: Char, rangeEnd: Char, step: Int): CharProgression
= CharProgression(rangeStart, rangeEnd, step)\r\n    }\r\n}\r\n\r\n/**\r\n * A progression of values of type `Int`.\r\n
*/\r\npublic open class IntProgression\r\n    internal constructor(\r\n        (\r\n            start: Int,\r\n\r\n
endInclusive:
Int,\r\n            step: Int\r\n        ) : Iterable<Int> {\r\n    init {\r\n        if (step == 0) throw
kotlin.IllegalArgumentException("Step must be non-zero.")\r\n        if (step == Int.MIN_VALUE) throw
kotlin.IllegalArgumentException("Step must be greater than Int.MIN_VALUE to avoid overflow on
negation.")\r\n    }\r\n\r\n    /**\r\n     * The first element in the progression.\r\n     */\r\n    public val first: Int =
start\r\n\r\n    /**\r\n     * The last element in the progression.\r\n     */\r\n    public val last: Int =
getProgressionLastElement(start.toInt(), endInclusive.toInt(), step).toInt()\r\n\r\n    /**\r\n     * The step of the
progression.\r\n     */\r\n    public val step: Int = step\r\n\r\n    override fun iterator(): IntIterator =
IntProgressionIterator(first, last, step)\r\n\r\n    /** Checks if the progression is empty. */\r\n    public open fun
isEmpty(): Boolean = if (step > 0) first > last else first < last\r\n\r\n    override fun equals(other: Any?): Boolean
= \r\n        other is IntProgression && (isEmpty() && other.isEmpty()) ||\r\n        first == other.first && last ==
other.last && step == other.step)\r\n\r\n    override fun hashCode(): Int = \r\n        if (isEmpty()) -1 else (31 * (31 *
first + last) + step)\r\n\r\n    override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first
downTo $last step ${-step}"\r\n\r\n    companion object {\r\n        /**\r\n         * Creates IntProgression within the
specified bounds of a closed range.\r\n         * The progression starts with the [rangeStart] value and goes toward
the [rangeEnd] value not excluding it, with the specified [step].\r\n         * In order to go backwards the [step] must

```


parameter declaration in the source code, `*` such as an instance, an extension receiver parameter or a value parameter.

```

public enum class Kind {
    /** Instance required to make a call to the member, or an
    outer class instance for an inner class constructor.
    INSTANCE,
    /** Extension receiver of an
    extension function or property.
    EXTENSION_RECEIVER,
    /** Ordinary named value
    parameter.
    VALUE,
}

```

`VALUE` `*` `true` if this parameter is optional and can be omitted when making a call via `[KCallable.callBy]`, or `false` otherwise.

`*` A parameter is optional in any of the two cases:

1. The default value is provided at the declaration of this parameter.
2. The parameter is declared in a member function and one of the corresponding parameters in the super functions is optional.

```

public val isOptional: Boolean

```

`*` `true` if this parameter is `vararg`.

See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/functions.html#variable-number-of-arguments-varargs) for more information.

```

@SinceKotlin("1.1")
public val isVararg: Boolean

```

Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

```

package kotlin.reflect

```

Represents a type. Type is usually either a class with optional type arguments, or a type parameter of some declaration, plus nullability.

```

public interface KType : KAnnotatedElement {

```

The declaration of the classifier used in this type.

For example, in the type `List<String>` the classifier would be the `[KClass]` instance for `[List]`.

Returns `null` if this type is not denotable in Kotlin, for example if it is an intersection type.

```

@SinceKotlin("1.1")
public val classifier: KClassifier?

```

Type arguments passed for the parameters of the classifier in this type.

For example, in the type `Array<out Number>` the only type argument is `out Number`.

In case this type is based on an inner class, the returned list contains the type arguments provided for the innermost class first, then its outer class, and so on.

For example, in the type `Outer<A, B>.Inner<C, D>` the returned list is `[C, D, A, B]`.

```

@SinceKotlin("1.1")
public val arguments: List<KTypeProjection>

```

`true` if this type was marked nullable in the source code.

For Kotlin types, it means that `null` value is allowed to be represented by this type.

In practice it means that the type was declared with a question mark at the end.

For non-Kotlin types, it means the type or the symbol which was declared with this type is annotated with a runtime-retained nullability annotation such as `[javax.annotation.Nullable]`.

Note that even if `[isMarkedNullable]` is false, values of the type can still be `null`.

This may happen if it is a type of the type parameter with a nullable upper bound:

```

fun <T> foo(t: T) {
    // isMarkedNullable == false for t's
    type, but t can be null here when T = "Any?"
}

```

`isMarkedNullable`:

```

Boolean

```

Represents a type projection. Type projection is usually the argument to another type in a type usage.

For example, in the type `Array<out Number>`, `out Number` is the covariant projection of the type represented by the class `Number`.

Type projection is either the star projection, or an entity consisting of a specific type plus optional variance.

See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/generics.html#type-projections) for more information.

```

@SinceKotlin("1.1")
public data class KTypeProjection constructor(
    /** The use-site
    variance specified in the projection, or `null` if this is a star projection.
    variance:
    KVariance?,
    /** The type specified in the projection, or `null` if this is a star projection.
    type: KType?
) {
    public companion object {
        /** Star projection,
        denoted by the `*` character.
        For example, in the type `KClass<*>`, `*` is the star projection.
        See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/generics.html#star-projections)
        for more information.
        STAR: KTypeProjection = KTypeProjection(null,
        null)
        /** Creates an invariant projection of a given type. Invariant projection is just the type
        itself,
        without any use-site variance modifiers applied to it.
        For example, in the type
        `Set<String>`, `String` is an invariant projection of the type represented by the class `String`.
        public fun invariant(type: KType): KTypeProjection =
            KTypeProjection(KVariance.INVARIANT,
            type)
        /** Creates a contravariant projection of a given type, denoted by the `in` modifier

```

applied to a type.
 * For example, in the type `MutableList<in Number>`, `in Number` is a contravariant projection of the type of class `Number`.
`public fun contravariant(type: KType): KTypeProjection =`
`KTypeProjection(KVariance.IN, type)`
 * Creates a covariant projection of a given type, denoted by the `out` modifier applied to a type.
 * For example, in the type `Array<out Number>`, `out Number` is a covariant projection of the type of class `Number`.
`public fun covariant(type: KType): KTypeProjection =`
`KTypeProjection(KVariance.OUT, type)`
 * Copyright 2010-2016 JetBrains s.r.o.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and limitations under the License.
`package kotlin.reflect`
`/**`
 `* Represents variance applied to a type parameter on the declaration site (*declaration-site variance*),`
 `* or to a type in a projection (*use-site variance*).`
 `* See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/generics.html#variance)`
 `* for more information.`
 `* @see [KTypeParameter.variance]`
 `* @see [KTypeProjection]`
 `*/`
`@SinceKotlin("1.1")`
`enum class KVariance {`
 `/**`
 `* The affected type parameter or type is *invariant*, which means it has no variance applied to it.`
 `*/`
 `INvariant, //`
 `/**`
 `* The affected type parameter or type is *contravariant*. Denoted by the `in` modifier in the source code.`
 `*/`
 `INvariant, //`
 `/**`
 `* The affected type parameter or type is *covariant*. Denoted by the `out` modifier in the source code.`
 `*/`
 `OUTvariant, //`
 * Copyright 2010-2016 JetBrains s.r.o.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and limitations under the License.
`package kotlin.reflect`
`/**`
 `* Visibility is an aspect of a Kotlin declaration regulating where that declaration is accessible in the source code.`
 `* Visibility can be changed with one of the following modifiers: `public`, `protected`, `internal`, `private`.`
 `* Note that some Java visibilities such as package-private and protected (which also gives access to items from the same package)`
 `* cannot be represented in Kotlin, so there's no [KVisibility] value corresponding to them.`
 `* See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/visibility-modifiers.html)`
 `* for more information.`
 `*/`
`@SinceKotlin("1.1")`
`enum class KVisibility {`
 `/**`
 `* Visibility of declarations marked with the `public` modifier, or with no modifier at all.`
 `*/`
 `PUBLIC, //`
 `/**`
 `* Visibility of declarations marked with the `protected` modifier.`
 `*/`
 `PROTECTED, //`
 `/**`
 `* Visibility of declarations marked with the `internal` modifier.`
 `*/`
 `INTERNAL, //`
 `/**`
 `* Visibility of declarations marked with the `private` modifier.`
 `*/`
 `PRIVATE, //`
 * Copyright 2010-2015 JetBrains s.r.o.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and limitations under the License.
`package kotlin`
`/**`
 `* The type with only one value: the `Unit` object. This type corresponds to the `void` type in Java.`
 `*/`
`public object Unit {`
 `override fun toString() = "kotlin.Unit"`
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 `*/`
`@JsName("arrayIterator")`
`internal fun arrayIterator(array: dynamic, type: String?) = when (type) {`
 `null -> {`
 `val arr: Array<dynamic> = array`
 `object :`

```

Iterator<dynamic> {\r\n      var index = 0\r\n      override fun hasNext() = index < arr.size\r\n      override
fun next() = if (index < arr.size) arr[index++] else throw NoSuchElementException("$index")\r\n    }\r\n  }\r\n
\r\n  \"BooleanArray\" -> booleanArrayIterator(array)\r\n  \"ByteArray\" -> byteArrayIterator(array)\r\n
\r\n  \"ShortArray\" -> shortArrayIterator(array)\r\n  \"CharArray\" -> charArrayIterator(array)\r\n  \"IntArray\" ->
intArrayIterator(array)\r\n  \"LongArray\" -> longArrayIterator(array)\r\n  \"FloatArray\" ->
floatArrayIterator(array)\r\n  \"DoubleArray\" -> doubleArrayIterator(array)\r\n  else -> throw
IllegalStateException(\"Unsupported type argument for arrayIterator:
$type\")\r\n}\r\n\r\n@JsName(\"booleanArrayIterator\")\r\ninternal fun booleanArrayIterator(array: BooleanArray)
= object : BooleanIterator() {\r\n  var index = 0\r\n  override fun hasNext() = index < array.size\r\n  override fun
nextBoolean() = if (index < array.size) array[index++] else throw
NoSuchElementException(\"$index\")\r\n}\r\n\r\n@JsName(\"byteArrayIterator\")\r\ninternal fun
byteArrayIterator(array: ByteArray) = object : ByteIterator() {\r\n  var index = 0\r\n  override fun hasNext() =
index < array.size\r\n  override fun nextByte() = if (index < array.size) array[index++] else throw
NoSuchElementException(\"$index\")\r\n}\r\n\r\n@JsName(\"shortArrayIterator\")\r\ninternal fun
shortArrayIterator(array: ShortArray) = object : ShortIterator() {\r\n  var index = 0\r\n  override fun hasNext() =
index < array.size\r\n  override fun nextShort() = if (index < array.size) array[index++] else throw
NoSuchElementException(\"$index\")\r\n}\r\n\r\n@JsName(\"charArrayIterator\")\r\ninternal fun
charArrayIterator(array: CharArray) = object : CharIterator() {\r\n  var index = 0\r\n  override fun hasNext() =
index < array.size\r\n  override fun nextChar() = if (index < array.size) array[index++] else throw
NoSuchElementException(\"$index\")\r\n}\r\n\r\n@JsName(\"intArrayIterator\")\r\ninternal fun
intArrayIterator(array: IntArray) = object : IntIterator() {\r\n  var index = 0\r\n  override fun hasNext() = index <
array.size\r\n  override fun nextInt() = if (index < array.size) array[index++] else throw
NoSuchElementException(\"$index\")\r\n}\r\n\r\n@JsName(\"floatArrayIterator\")\r\ninternal fun
floatArrayIterator(array: FloatArray) = object : FloatIterator() {\r\n  var index = 0\r\n  override fun hasNext() =
index < array.size\r\n  override fun nextFloat() = if (index < array.size) array[index++] else throw
NoSuchElementException(\"$index\")\r\n}\r\n\r\n@JsName(\"doubleArrayIterator\")\r\ninternal fun
doubleArrayIterator(array: DoubleArray) = object : DoubleIterator() {\r\n  var index = 0\r\n  override fun
hasNext() = index < array.size\r\n  override fun nextDouble() = if (index < array.size) array[index++] else throw
NoSuchElementException(\"$index\")\r\n}\r\n\r\n@JsName(\"longArrayIterator\")\r\ninternal fun
longArrayIterator(array: LongArray) = object : LongIterator() {\r\n  var index = 0\r\n  override fun hasNext() =
index < array.size\r\n  override fun nextLong() = if (index < array.size) array[index++] else throw
NoSuchElementException(\"$index\")\r\n}\r\n\r\n@JsName(\"PropertyMetadata\")\r\ninternal class
PropertyMetadata(@JsName(\"callableName\") val name:
String)\r\n\r\n@JsName(\"noWhenBranchMatched\")\r\ninternal fun noWhenBranchMatched(): Nothing = throw
NoWhenBranchMatchedException()\r\n\r\n@JsName(\"subSequence\")\r\ninternal fun subSequence(c:
CharSequence, startIndex: Int, endIndex: Int): CharSequence {\r\n  if (c is String) {\r\n    return
c.substring(startIndex, endIndex)\r\n  } else {\r\n    return c.asDynamic().`subSequence_vux9f0$`(startIndex,
endIndex)\r\n  }\r\n}\r\n\r\n@JsName(\"captureStack\")\r\ninternal fun captureStack(baseClass: JsClass<in
Throwable>, instance: Throwable) {\r\n  if (js(\"Error\").captureStackTrace) {\r\n
js(\"Error\").captureStackTrace(instance, instance::class.js);\r\n  } else {\r\n    instance.asDynamic().stack =
js(\"new Error()\").stack;\r\n  }\r\n}\r\n\r\n@JsName(\"newThrowable\")\r\ninternal fun newThrowable(message:
String?, cause: Throwable?): Throwable {\r\n  val throwable = js(\"new Error()\")\r\n  throwable.message = if
(jsTypeOf(message) == \"undefined\") {\r\n    if (cause != null) cause.toString() else null\r\n  } else {\r\n
message\r\n  }\r\n  throwable.cause = cause\r\n  throwable.name = \"Throwable\"\r\n  return
throwable\r\n}\r\n\r\n@JsName(\"BoxedChar\")\r\ninternal class BoxedChar(val c: Int) : Comparable<Int> {\r\n
override fun equals(other: Any?): Boolean {\r\n  return other is BoxedChar && c == other.c\r\n  }\r\n\r\n
override fun hashCode(): Int {\r\n  return c\r\n  }\r\n\r\n  override fun toString(): String {\r\n  return
js(\"this.c\").unsafeCast<Char>().toString()\r\n  }\r\n\r\n  override fun compareTo(other: Int): Int {\r\n  return

```

```

js("this.c - other").unsafeCast<Int>()\r\n } \r\n\r\n @JsName("valueOf")\r\n public fun valueOf(): Int {\r\n
return c\r\n } \r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\ninternal inline fun <T> concat(args: Array<T>): T {\r\n
val typed = js("Array")(args.size)\r\n for (i in args.indices) {\r\n val arr = args[i]\r\n if (arr !is Array<*>)\r\n
{\r\n typed[i] = js("[]").slice.call(arr)\r\n } else {\r\n typed[i] = arr\r\n }\r\n }\r\n return
js("[]").concat.apply(js("[]"), typed);\r\n}\r\n\r\n\r\n** Concat regular Array's and TypedArray's into an Array.\r\n
*\r\n\r\n@PublishedApi\r\n\r\n@JsName("arrayConcat")\r\n\r\n@Suppress("UNUSED_PARAMETER")\r\n\r\ninternal fun
<T> arrayConcat(a: T, b: T): T {\r\n return concat(js("arguments"))\r\n}\r\n\r\n\r\n** Concat primitive arrays. Main
use: prepare vararg arguments.\r\n\r\n* For compatibility with 1.1.0 the arguments may be a mixture of Array's and
TypedArray's.\r\n\r\n*\r\n\r\n* If the first argument is TypedArray (Byte-, Short-, Char-, Int-, Float-, and DoubleArray)
returns a TypedArray, otherwise an Array.\r\n\r\n* If the first argument has the $type$ property (Boolean-, Char-, and
LongArray) copy its value to result.$type$.\r\n\r\n* If the first argument is a regular Array without the $type$ property
default to arrayConcat.\r\n
*\r\n\r\n@PublishedApi\r\n\r\n@JsName("primitiveArrayConcat")\r\n\r\n@Suppress("UNUSED_PARAMETER")\r\n\r\ninter
nal fun <T> primitiveArrayConcat(a: T, b: T): T {\r\n val args: Array<T> = js("arguments")\r\n if (a is
Array<*> && a.asDynamic().`$type$` === undefined) {\r\n return concat(args)\r\n } else {\r\n var size =
0\r\n for (i in args.indices) {\r\n size += args[i].asDynamic().length as Int\r\n }\r\n val result =
js("new a.constructor(size)")\r\n kotlin.copyArrayType(a, result)\r\n size = 0\r\n for (i in args.indices)
{\r\n val arr = args[i].asDynamic()\r\n for (j in 0 until arr.length) {\r\n result[size++] =
arr[j]\r\n }\r\n }\r\n return result\r\n }\r\n}\r\n\r\n\r\n@JsName("booleanArrayOf")\r\n\r\ninternal fun
booleanArrayOf() = withType("BooleanArray", js("[]").slice.call(arguments))\r\n\r\n\r\n@JsName("charArrayOf")
// The arguments have to be slice'd here because of Rhino (see KT-16974)\r\n\r\ninternal fun charArrayOf() =
withType("CharArray", js("new
Uint16Array([]).slice.call(arguments)))\r\n\r\n\r\n@JsName("longArrayOf")\r\n\r\ninternal fun longArrayOf() =
withType("LongArray",
js("[]").slice.call(arguments))\r\n}\r\n\r\n\r\n@JsName("withType")\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\ninternal inline fun
withType(type: String, array: dynamic): dynamic {\r\n array.`$type$` = type\r\n return array\r\n}\r\n\r\n\r\n*
Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n\r\n* Use of this source code
is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n\r\n*\r\n\r\n\r\n\r\npackage
kotlin.js\r\n\r\n\r\n**\r\n\r\n* Function corresponding to JavaScript's `typeof` operator\r\n
*\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\n@Suppress("UNUSED_PARAMETER")\r\n\r\npublic inline fun jsTypeOf(a:
Any?): String = js("typeof a")\r\n\r\n\r\n\r\n**\r\n\r\n* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.\r\n\r\n* Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n\r\n*\r\n\r\n\r\n\r\n@file:Suppress("UNUSED_PARAMETER",
"NOTHING_TO_INLINE")\r\n\r\n\r\npackage kotlin\r\n\r\n\r\n**\r\n\r\n* Returns an empty array of the specified type
[T].\r\n\r\n*\r\n\r\n\r\npublic inline fun <T> emptyArray(): Array<T> = js("[]")\r\n\r\n\r\n@library\r\n\r\n\r\npublic fun <T>
arrayOf(vararg elements: T): Array<T> = definedExternally\r\n\r\n\r\n@library\r\n\r\n\r\npublic fun doubleArrayOf(vararg
elements: Double): DoubleArray = definedExternally\r\n\r\n\r\n@library\r\n\r\n\r\npublic fun floatArrayOf(vararg elements:
Float): FloatArray = definedExternally\r\n\r\n\r\n@library\r\n\r\n\r\npublic fun longArrayOf(vararg elements: Long):
LongArray = definedExternally\r\n\r\n\r\n@library\r\n\r\n\r\npublic fun intArrayOf(vararg elements: Int): IntArray =
definedExternally\r\n\r\n\r\n@library\r\n\r\n\r\npublic fun charArrayOf(vararg elements: Char): CharArray =
definedExternally\r\n\r\n\r\n@library\r\n\r\n\r\npublic fun shortArrayOf(vararg elements: Short): ShortArray =
definedExternally\r\n\r\n\r\n@library\r\n\r\n\r\npublic fun byteArrayOf(vararg elements: Byte): ByteArray =
definedExternally\r\n\r\n\r\n@library\r\n\r\n\r\npublic fun booleanArrayOf(vararg elements: Boolean): BooleanArray =
definedExternally\r\n\r\n\r\n\r\n**\r\n\r\n* Creates a new instance of the [Lazy] that uses the specified initialization
function [initializer].\r\n\r\n*\r\n\r\n\r\npublic actual fun <T> lazy(initializer: () -> T): Lazy<T> =
UnsafeLazyImpl(initializer)\r\n\r\n\r\n\r\n**\r\n\r\n* Creates a new instance of the [Lazy] that uses the specified
initialization function [initializer].\r\n\r\n*\r\n\r\n\r\n* The [mode] parameter is ignored. \r\n\r\n\r\npublic actual fun <T> lazy(mode:
LazyThreadSafetyMode, initializer: () -> T): Lazy<T> = UnsafeLazyImpl(initializer)\r\n\r\n\r\n\r\n**\r\n\r\n* Creates a new

```



```

instance of the [Lazy] that uses the specified initialization function [initializer].\r\n * The [lock] parameter is ignored.\r\n */\r\n\npublic actual fun <T> lazy(lock: Any?, initializer: () -> T): Lazy<T> =
UnsafeLazyImpl(initializer)\r\n\r\n\r\ninternal fun fillFrom(src: dynamic, dst: dynamic): dynamic {\r\n    val srcLen: Int = src.length\r\n    val dstLen: Int = dst.length\r\n    var index: Int = 0\r\n    while (index < srcLen && index < dstLen) dst[index] = src[index++]\r\n    return dst\r\n}\r\n\r\n\r\ninternal fun arrayCopyResize(source: dynamic, newSize: Int, defaultValue: Any?): dynamic {\r\n    val result = source.slice(0, newSize)\r\n    copyArrayType(source, result)\r\n    var index: Int = source.length\r\n    if (newSize > index) {\r\n        result.length = newSize\r\n        while (index < newSize) result[index++] = defaultValue\r\n    }\r\n    return result\r\n}\r\n\r\n\r\ninternal fun <T> arrayPlusCollection(array: dynamic, collection: Collection<T>): dynamic {\r\n    val result = array.slice()\r\n    result.length += collection.size\r\n    copyArrayType(array, result)\r\n    var index: Int = array.length\r\n    for (element in collection) result[index++] = element\r\n    return result\r\n}\r\n\r\n\r\ninternal fun <T> fillFromCollection(dst: dynamic, startIndex: Int, collection: Collection<T>): dynamic {\r\n    var index = startIndex\r\n    for (element in collection) dst[index++] = element\r\n    return dst\r\n}\r\n\r\n\r\ninternal inline fun copyArrayType(from: dynamic, to: dynamic) {\r\n    if (from.`$type$` !== undefined) {\r\n        to.`$type$` = from.`$type$`\r\n    }\r\n}\r\n\r\n\r\ninternal inline fun jsIsType(obj: dynamic, jsClass: dynamic) = js("Kotlin").isType(obj, jsClass), "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n\r\npackage kotlin.coroutines\r\n\r\n\r\nimport kotlin.coroutines.intrinsics.COROUTINE_SUSPENDED\r\n\r\n\r\n@SinceKotlin("1.3")\r\n@JsName("CoroutineImpl")\r\ninternal abstract class CoroutineImpl(private val resultContinuation: Continuation<Any?>) : Continuation<Any?> {\r\n    protected var state = 0\r\n    protected var exceptionState = 0\r\n    protected var result: Any? = null\r\n    protected var exception: Throwable? = null\r\n    protected var finallyPath: Array<Int?> = null\r\n\r\n    public override val context: CoroutineContext = resultContinuation.context\r\n\r\n    private var intercepted_: Continuation<Any?>? = null\r\n\r\n    public fun intercepted(): Continuation<Any?> =\r\n        intercepted_ ?: (context[ContinuationInterceptor]?.interceptContinuation(this) ?: this)\r\n\r\n    .also {\r\n        intercepted_ = it\r\n    }\r\n\r\n    override fun resumeWith(result: Result<Any?>) {\r\n        var current = this\r\n        var currentResult: Any? = result.getOrNull()\r\n        var currentException: Throwable? = result.exceptionOrNull()\r\n\r\n        // This loop unrolls recursion in current.resumeWith(param) to make saner and shorter stack traces on resume\r\n        while (true) {\r\n            with(current) {\r\n                val completion = resultContinuation\r\n\r\n                // Set result and exception fields in the current continuation\r\n                if (currentException == null) {\r\n                    this.result = currentResult\r\n                } else {\r\n                    state = exceptionState\r\n                    exception = currentException\r\n                }\r\n\r\n                try {\r\n                    val outcome = doResume()\r\n                    if (outcome === COROUTINE_SUSPENDED) return\r\n                    currentResult = outcome\r\n                    currentException = null\r\n                } catch (exception: dynamic) { // Catch all exceptions\r\n                    currentResult = null\r\n                    currentException = exception.unsafeCast<Throwable>()\r\n                }\r\n\r\n                releaseIntercepted() // this state machine instance is terminating\r\n\r\n                if (completion is CoroutineImpl) {\r\n                    // unrolling recursion via loop\r\n                    current = completion\r\n                } else {\r\n                    // top-level completion reached -- invoke and return\r\n                    currentException?.let {\r\n                        completion.resumeWithException(it)\r\n                    } ?: completion.resume(currentResult)\r\n\r\n                    return\r\n                }\r\n            }\r\n\r\n            }\r\n\r\n        private fun releaseIntercepted() {\r\n            val intercepted = intercepted_ \r\n            if (intercepted != null && intercepted !== this) {\r\n                context[ContinuationInterceptor]!!.releaseInterceptedContinuation(intercepted)\r\n            }\r\n            this.intercepted_ = CompletedContinuation // just in case\r\n        }\r\n\r\n        protected abstract fun doResume(): Any?\r\n\r\n    internal object CompletedContinuation : Continuation<Any?> {\r\n        override val context: CoroutineContext\r\n        get() = error("This continuation is already complete")\r\n\r\n        override fun resumeWith(result: Result<Any?>) {\r\n            error("This continuation is already complete")\r\n        }\r\n\r\n        override fun toString(): String = "This continuation is already complete"\r\n    }\r\n\r\n    /*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the

```

```

Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*\r\n\r\n@file:Suppress("\r\nUNCHECKED_CAST", "RedundantVisibilityModifier")\r\n\r\n\r\npackage
kotlin\r\n\r\n\r\nimport kotlin.contracts.*\r\n\r\nimport kotlin.internal.InlineOnly\r\n\r\nimport
kotlin.jvm.JvmField\r\n\r\n\r\n/**\r\n * A discriminated union that encapsulates successful outcome with a value of
type [T]\r\n * or a failure with an arbitrary [Throwable] exception.\r\n\r\n
*\r\n\r\n@Suppress("\r\nNON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")\r\n\r\n@SinceKotlin("1.3\r\n")\r\n\r\npublic inline class Result<out T> @PublishedApi internal constructor(\r\n
@PublishedApi\r\n internal val
value: Any?\r\n) : Serializable {\r\n // discovery\r\n\r\n // **\r\n * Returns `true` if this instance represents
successful outcome.\r\n * In this case [isFailure] returns `false`.\r\n * \r\n public val isSuccess: Boolean get()
= value !is Failure\r\n\r\n // **\r\n * Returns `true` if this instance represents failed outcome.\r\n * In this case
[isSuccess] returns `false`.\r\n * \r\n public val isFailure: Boolean get() = value is Failure\r\n\r\n // value &
exception retrieval\r\n\r\n // **\r\n * Returns the encapsulated value if this instance represents
[success][Result.isSuccess] or `null`\r\n * if it is [failure][Result.isFailure].\r\n * \r\n * This function is
shorthand for `getOrElse { null }` (see [getOrElse]) or\r\n * `fold(onSuccess = { it }, onFailure = { null })` (see
[fold]).\r\n * \r\n @InlineOnly\r\n public inline fun getOrNull(): T? =\r\n when {\r\n isFailure ->
null\r\n else -> value as T\r\n }\r\n\r\n // **\r\n * Returns the encapsulated exception if this instance
represents [failure][isFailure] or `null`\r\n * if it is [success][isSuccess].\r\n * \r\n * This function is shorthand
for `fold(onSuccess = { null }, onFailure = { it })` (see [fold]).\r\n * \r\n public fun exceptionOrNull():
Throwable? =\r\n when (value) {\r\n is Failure -> value.exception\r\n else -> null\r\n }\r\n\r\n // **\r\n * Returns a string `Success(v)` if this instance represents [success][Result.isSuccess]\r\n * where `v` is
a string representation of the value or a string `Failure(x)` if\r\n * it is [failure][isFailure] where `x` is a string
representation of the exception.\r\n * \r\n public override fun toString(): String =\r\n when (value) {\r\n
is Failure -> value.toString() // "Failure($exception)"
\r\n else -> "Success($value)"
\r\n }\r\n\r\n //
companion with constructors\r\n\r\n // **\r\n * Companion object for [Result] class that contains its constructor
functions\r\n * [success] and [failure].\r\n * \r\n public companion object {\r\n // **\r\n * Returns an
instance that encapsulates the given [value] as successful value.\r\n * \r\n @InlineOnly\r\n public
inline fun <T> success(value: T): Result<T> =\r\n Result(value)\r\n\r\n // **\r\n * Returns an instance
that encapsulates the given [exception] as failure.\r\n * \r\n @InlineOnly\r\n public inline fun <T>
failure(exception: Throwable): Result<T> =\r\n Result(createFailure(exception))\r\n }\r\n\r\n internal
class Failure(\r\n @JvmField\r\n val exception: Throwable\r\n) : Serializable {\r\n override fun
equals(other: Any?): Boolean = other is Failure && exception == other.exception\r\n override fun hashCode():
Int = exception.hashCode()\r\n override fun toString(): String = "Failure($exception)"
\r\n}\r\n\r\n\r\n/**\r\n * Creates an instance of internal marker [Result.Failure] class to\r\n * make sure that this class is
not exposed in ABI.\r\n * \r\n @PublishedApi\r\n @SinceKotlin("1.3")\r\n internal fun createFailure(exception:
Throwable): Any =\r\n Result.Failure(exception)\r\n\r\n\r\n/**\r\n * Throws exception if the result is failure. This
internal function minimizes\r\n * inlined bytecode for [getOrThrow] and makes sure that in the future we can\r\n *
add some exception-augmenting logic here (if needed).\r\n\r\n
*\r\n\r\n@PublishedApi\r\n@SinceKotlin("1.3")\r\n internal fun Result<*>.throwOnFailure() {\r\n if (value is
Result.Failure) throw value.exception\r\n}\r\n\r\n\r\n/**\r\n * Calls the specified function [block] and returns its
encapsulated result if invocation was successful,\r\n * catching and encapsulating any thrown exception as a
failure.\r\n * \r\n @InlineOnly\r\n @SinceKotlin("1.3")\r\n public inline fun <R> runCatching(block: () -> R):
Result<R> {\r\n return try {\r\n Result.success(block())\r\n } catch (e: Throwable) {\r\n
Result.failure(e)\r\n }\r\n}\r\n\r\n\r\n/**\r\n * Calls the specified function [block] with `this` value as its receiver and
returns its encapsulated result\r\n * if invocation was successful, catching and encapsulating any thrown exception
as a failure.\r\n * \r\n @InlineOnly\r\n @SinceKotlin("1.3")\r\n public inline fun <T, R> T.runCatching(block: T.() -
> R): Result<R> {\r\n return try {\r\n Result.success(block())\r\n } catch (e: Throwable) {\r\n
Result.failure(e)\r\n }\r\n}\r\n\r\n\r\n// -- extensions ---\r\n\r\n\r\n/**\r\n * Returns the encapsulated value if this instance
represents [success][Result.isSuccess] or throws the encapsulated exception\r\n * if it is

```

[failure][Result.isFailure].\r\n * This function is shorthand for `getOrElse { throw it }` (see [getOrElse]).\r\n * \r\n @InlineOnly\r\n @SinceKotlin("1.3")\r\n public inline fun <T> Result<T>.getOrThrow(): T {\r\n throwOnFailure()\r\n return value as T\r\n }\r\n /**\r\n * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or the\r\n * result of [onFailure] function for encapsulated exception if it is [failure][Result.isFailure].\r\n * Note, that an exception thrown by [onFailure] function is rethrown by this function.\r\n * This function is shorthand for `fold(onSuccess = { it }, onFailure = onFailure)` (see [fold]).\r\n * \r\n @InlineOnly\r\n @SinceKotlin("1.3")\r\n public inline fun <R, T : R> Result<T>.getOrElse(onFailure: (exception: Throwable) -> R): R {\r\n contract {\r\n callsInPlace(onFailure, InvocationKind.AT_MOST_ONCE)\r\n }\r\n return when (val exception = exceptionOrNull()) {\r\n null -> value as T\r\n else -> onFailure(exception)\r\n }\r\n }\r\n /**\r\n * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or the\r\n * [defaultValue] if it is [failure][Result.isFailure].\r\n * This function is shorthand for `getOrElse { defaultValue }` (see [getOrElse]).\r\n * \r\n @InlineOnly\r\n @SinceKotlin("1.3")\r\n public inline fun <R, T : R> Result<T>.getOrDefault(defaultValue: R): R {\r\n if (isFailure) return defaultValue\r\n return value as T\r\n }\r\n /**\r\n * Returns the the result of [onSuccess] for encapsulated value if this instance represents [success][Result.isSuccess]\r\n * or the result of [onFailure] function for encapsulated exception if it is [failure][Result.isFailure].\r\n * Note, that an exception thrown by [onSuccess] or by [onFailure] function is rethrown by this function.\r\n * \r\n @InlineOnly\r\n @SinceKotlin("1.3")\r\n public inline fun <R, T> Result<T>.fold(\r\n onSuccess: (value: T) -> R,\r\n onFailure: (exception: Throwable) -> R\r\n): R {\r\n contract {\r\n callsInPlace(onSuccess, InvocationKind.AT_MOST_ONCE)\r\n callsInPlace(onFailure, InvocationKind.AT_MOST_ONCE)\r\n }\r\n return when (val exception = exceptionOrNull()) {\r\n null -> onSuccess(value as T)\r\n else -> onFailure(exception)\r\n }\r\n }\r\n /**\r\n * Returns the encapsulated result of the given [transform] function applied to encapsulated value\r\n * if this instance represents [success][Result.isSuccess] or the\r\n * original encapsulated exception if it is [failure][Result.isFailure].\r\n * Note, that an exception thrown by [transform] function is rethrown by this function.\r\n * See [mapCatching] for an alternative that encapsulates exceptions.\r\n * \r\n @InlineOnly\r\n @SinceKotlin("1.3")\r\n public inline fun <R, T> Result<T>.map(transform: (value: T) -> R): Result<R> {\r\n contract {\r\n callsInPlace(transform, InvocationKind.AT_MOST_ONCE)\r\n }\r\n return when {\r\n isSuccess -> Result.success(transform(value as T))\r\n else -> Result(value)\r\n }\r\n }\r\n /**\r\n * Returns the encapsulated result of the given [transform] function applied to encapsulated value\r\n * if this instance represents [success][Result.isSuccess] or the\r\n * original encapsulated exception if it is [failure][Result.isFailure].\r\n * Any exception thrown by [transform] function is caught, encapsulated as a failure and returned by this function.\r\n * See [map] for an alternative that rethrows exceptions.\r\n * \r\n @InlineOnly\r\n @SinceKotlin("1.3")\r\n public inline fun <R, T> Result<T>.mapCatching(transform: (value: T) -> R): Result<R> {\r\n return when {\r\n isSuccess -> runCatching { transform(value as T) }\r\n else -> Result(value)\r\n }\r\n }\r\n /**\r\n * Returns the encapsulated result of the given [transform] function applied to encapsulated exception\r\n * if this instance represents [failure][Result.isFailure] or the\r\n * original encapsulated value if it is [success][Result.isSuccess].\r\n * Note, that an exception thrown by [transform] function is rethrown by this function.\r\n * See [recoverCatching] for an alternative that encapsulates exceptions.\r\n * \r\n @InlineOnly\r\n @SinceKotlin("1.3")\r\n public inline fun <R, T : R> Result<T>.recover(transform: (exception: Throwable) -> R): Result<R> {\r\n contract {\r\n callsInPlace(transform, InvocationKind.AT_MOST_ONCE)\r\n }\r\n return when (val exception = exceptionOrNull()) {\r\n null -> this\r\n else -> Result.success(transform(exception))\r\n }\r\n }\r\n /**\r\n * Returns the encapsulated result of the given [transform] function applied to encapsulated exception\r\n * if this instance represents [failure][Result.isFailure] or the\r\n * original encapsulated value if it is [success][Result.isSuccess].\r\n * Any exception thrown by [transform] function is caught, encapsulated as a failure and returned by this function.\r\n * See [recover] for an alternative that rethrows exceptions.\r\n * \r\n @InlineOnly\r\n @SinceKotlin("1.3")\r\n public inline fun <R, T : R> Result<T>.recoverCatching(transform: (exception: Throwable) -> R): Result<R> {\r\n val

invocation of any resume function on the resulting continuation will produce an [IllegalStateException].

```

*^@SinceKotlin("1.3")@Suppress("UNCHECKED_CAST")\r\npublic fun <R, T> (suspend R.() ->
T).createCoroutine(\r\n receiver: R,\r\n completion: Continuation<T>)\r\n): Continuation<Unit> =\r\n
SafeContinuation(createCoroutineUnintercepted(receiver, completion).intercepted(),
COROUTINE_SUSPENDED)\r\n\r\n/**\r\n * Starts a coroutine without a receiver and with result type [T].\r\n *
This function creates and starts a new, fresh instance of suspendable computation every time it is invoked.\r\n * The
[completion] continuation is invoked when the coroutine completes with a result or an exception.\r\n
*^@SinceKotlin("1.3")@Suppress("UNCHECKED_CAST")\r\npublic fun <T> (suspend () ->
T).startCoroutine(\r\n completion: Continuation<T>)\r\n) {\r\n
createCoroutineUnintercepted(completion).intercepted().resume(Unit)\r\n}\r\n\r\n/**\r\n * Starts a coroutine with
receiver type [R] and result type [T].\r\n * This function creates and starts a new, fresh instance of suspendable
computation every time it is invoked.\r\n * The [completion] continuation is invoked when the coroutine completes
with a result or an exception.\r\n
*^@SinceKotlin("1.3")@Suppress("UNCHECKED_CAST")\r\npublic fun <R, T> (suspend R.() -> T).startCoroutine(\r\n receiver: R,\r\n completion: Continuation<T>)\r\n) {\r\n
createCoroutineUnintercepted(receiver, completion).intercepted().resume(Unit)\r\n}\r\n\r\n/**\r\n * Obtains the
current continuation instance inside suspend functions and suspends\r\n * the currently running coroutine.\r\n
*^@SinceKotlin("1.3")@Suppress("UNCHECKED_CAST")\r\npublic suspend inline fun
<T> suspendCoroutine(crossinline block: (Continuation<T>) -> Unit): T =\r\n
suspendCoroutineUninterceptedOrReturn { c: Continuation<T> ->\r\n val safe =
SafeContinuation(c.intercepted())\r\n block(safe)\r\n safe.getOrThrow()\r\n }\r\n\r\n/**\r\n * Returns the
context of the current coroutine.\r\n
*^@SinceKotlin("1.3")@Suppress("WRONG_MODIFIER_TARGET")\r\n@InlineOnly\r\npublic suspend
inline val coroutineContext: CoroutineContext\r\n get() {\r\n throw NotImplementedError("Implemented as
intrinsic")\r\n }\r\n" ,"/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n
*^@SinceKotlin("1.3")@Suppress("UNCHECKED_CAST")\r\npublic suspend inline fun
<T> (suspend () -> T).startCoroutineUninterceptedOrReturn(\r\n completion: Continuation<T>)\r\n): Any? =
this.asDynamic()(completion, false)\r\n\r\n/**\r\n * Starts an unintercepted coroutine with receiver type [R] and
result type [T] and executes it until its first suspension.\r\n * Returns the result of the coroutine or throws its
exception if it does not suspend or [COROUTINE_SUSPENDED] if it suspends.\r\n * In the latter case, the
[completion] continuation is invoked when the coroutine completes with a result or an exception.\r\n
*^@SinceKotlin("1.3")@Suppress("UNCHECKED_CAST")\r\npublic suspend inline fun
<R, T> (suspend R.() -> T).startCoroutineUninterceptedOrReturn(\r\n receiver: R,\r\n completion: Continuation<T>)\r\n): Any? =
this.asDynamic()(completion, receiver, false)\r\n\r\n/**\r\n * Starts an unintercepted coroutine without a
receiver and with result type [T] and executes it until its first suspension.\r\n * Returns the result of the
coroutine or throws its exception if it does not suspend or [COROUTINE_SUSPENDED] if it suspends.\r\n * In the
latter case, the [completion] continuation is invoked when the coroutine completes with a result or an exception.\r\n
*^@SinceKotlin("1.3")@Suppress("UNCHECKED_CAST")\r\npublic suspend inline fun
<T> (suspend () -> T).startCoroutineUninterceptedOrReturn(\r\n completion: Continuation<T>)\r\n): Any? =
this.asDynamic()(completion, false)\r\n

```

```

T> (suspend R.() -> T).startCoroutineUninterceptedOrReturn(\r\n receiver: R,\r\n completion:
Continuation<T>)\r\n): Any? = this.asDynamic()(receiver, completion, false)\r\n\r\n\r\n\r\n**\r\n * Creates
unintercepted coroutine without receiver and with result type [T].\r\n * This function creates a new, fresh instance of
suspendable computation every time it is invoked.\r\n * \r\n * To start executing the created coroutine, invoke
`resume(Unit)` on the returned [Continuation] instance.\r\n * The [completion] continuation is invoked when
coroutine completes with result or exception.\r\n * \r\n * This function returns unintercepted continuation.\r\n *
Invocation of `resume(Unit)` starts coroutine immediately in the invoker's call stack without going through the\r\n *
[ContinuationInterceptor] that might be present in the completion's [CoroutineContext].\r\n * It is the invoker's
responsibility to ensure that a proper invocation context is established.\r\n * Note that [completion] of this function
may get invoked in an arbitrary context.\r\n * \r\n * [Continuation.intercepted] can be used to acquire the intercepted
continuation.\r\n * Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of\r\n * both
the coroutine and [completion] happens in the invocation context established by\r\n * [ContinuationInterceptor].\r\n
*\r\n * Repeated invocation of any resume function on the resulting continuation corrupts the\r\n * state machine of
the coroutine and may result in arbitrary behaviour or exception.\r\n */\r\n@SinceKotlin("1.3")\r\npublic actual fun
<T> (suspend () -> T).createCoroutineUnintercepted(\r\n completion: Continuation<T>)\r\n): Continuation<Unit>
=\r\n // Kotlin/JS suspend lambdas have an extra parameter `suspended`\r\n if (this.asDynamic().length == 2)
{\r\n // When `suspended` is true the continuation is created, but not executed\r\n
this.asDynamic()(completion, true)\r\n } else {\r\n createCoroutineFromSuspendFunction(completion) {\r\n
this.asDynamic()(completion)\r\n }\r\n }\r\n\r\n**\r\n * Creates unintercepted coroutine with receiver
type [R] and result type [T].\r\n * This function creates a new, fresh instance of suspendable computation every time
it is invoked.\r\n * \r\n * To start executing the created coroutine, invoke `resume(Unit)` on the returned
[Continuation] instance.\r\n * The [completion] continuation is invoked when coroutine completes with result or
exception.\r\n * \r\n * This function returns unintercepted continuation.\r\n * Invocation of `resume(Unit)` starts
coroutine immediately in the invoker's call stack without going through the\r\n * [ContinuationInterceptor] that
might be present in the completion's [CoroutineContext].\r\n * It is the invoker's responsibility to ensure that a
proper invocation context is established.\r\n * Note that [completion] of this function may get invoked in an
arbitrary context.\r\n * \r\n * [Continuation.intercepted] can be used to acquire the intercepted continuation.\r\n *
Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of\r\n * both the coroutine and
[completion] happens in the invocation context established by\r\n * [ContinuationInterceptor].\r\n * \r\n * Repeated
invocation of any resume function on the resulting continuation corrupts the\r\n * state machine of the coroutine and
may result in arbitrary behaviour or exception.\r\n */\r\n@SinceKotlin("1.3")\r\npublic actual fun <R, T> (suspend
R.() -> T).createCoroutineUnintercepted(\r\n receiver: R,\r\n completion: Continuation<T>)\r\n):
Continuation<Unit> =\r\n // Kotlin/JS suspend lambdas have an extra parameter `suspended`\r\n if
(this.asDynamic().length == 3) {\r\n // When `suspended` is true the continuation is created, but not executed\r\n
this.asDynamic()(receiver, completion, true)\r\n } else {\r\n
createCoroutineFromSuspendFunction(completion) {\r\n this.asDynamic()(receiver, completion)\r\n }\r\n
}\r\n\r\n**\r\n * Intercepts this continuation with [ContinuationInterceptor].\r\n * \r\n * This function shall be used
on the immediate result of [createCoroutineUnintercepted] or [suspendCoroutineUninterceptedOrReturn],\r\n * in
which case it checks for [ContinuationInterceptor] in the continuation's [context][Continuation.context],\r\n *
invokes [ContinuationInterceptor.interceptContinuation], caches and returns the result.\r\n * \r\n * If this function is
invoked on other [Continuation] instances it returns `this` continuation unchanged.\r\n
*\r\n@SinceKotlin("1.3")\r\npublic actual fun <T> Continuation<T>.intercepted(): Continuation<T> =\r\n (this
as? CoroutineImpl)?.intercepted() ?: this\r\n\r\nprivate inline fun <T> createCoroutineFromSuspendFunction(\r\n
completion: Continuation<T>,\r\n crossinline block: () -> Any?)\r\n): Continuation<Unit> {\r\n
@Suppress("UNCHECKED_CAST")\r\n return object : CoroutineImpl(completion as Continuation<Any?>)\r\n
{\r\n override fun doResume(): Any? {\r\n exception?.let { throw it }\r\n return block()\r\n
}\r\n }\r\n }\r\n\r\n**\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the

```



```

JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n/\n//
NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n\nimport kotlin.js.*\nimport
kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Reverses elements in the list in-place.\n */\npublic
actual fun <T> MutableList<T>.reverse(): Unit {\n    val midPoint = (size / 2) - 1\n    if (midPoint < 0) return\n    var
reverseIndex = lastIndex\n    for (index in 0..midPoint) {\n        val tmp = this[index]\n        this[index] =
this[reverseIndex]\n        this[reverseIndex] = tmp\n        reverseIndex--\n    }\n}\n\n"/\n\n * Copyright 2010-2019
JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.text\n\n/\n// NOTE:
THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n\nimport kotlin.js.*\n\n/**\n * Returns a
character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char
sequence.\n */\n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic actual fun
CharSequence.elementAt(index: Int): Char {\n    return elementAtOrElse(index) { throw
IndexOutOfBoundsException("index: $index, length: $length") }\n}\n\n"/\n\n * Copyright 2010-2019 JetBrains
s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0
license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.collections\n\n/\n// NOTE: THIS
FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n\nimport kotlin.js.*\nimport
kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns an element at the given [index] or throws
an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n */\n * @sample
samples.collections.Collections.Elements.elementAt\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic actual fun
UIntArray.elementAt(index: Int):
UInt {\n    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }\n
}\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is
out of bounds of this array.\n */\n * @sample samples.collections.Collections.Elements.elementAt\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic actual fun
ULongArray.elementAt(index: Int):
ULong {\n    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }\n
}\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is
out of bounds of this array.\n */\n * @sample samples.collections.Collections.Elements.elementAt\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic actual fun
UByteArray.elementAt(index: Int):
UByte {\n    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }\n
}\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is
out of bounds of this array.\n */\n * @sample samples.collections.Collections.Elements.elementAt\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic actual fun
UShortArray.elementAt(index: Int):
UShort {\n    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }\n
}\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic actual fun
UIntArray.asList(): List<UInt> {\n
return object : AbstractList<UInt>(), RandomAccess {\n    override val size: Int get() = this@asList.size\n
override fun isEmpty(): Boolean = this@asList.isEmpty()\n    override fun contains(element: UInt): Boolean =
this@asList.contains(element)\n    override fun get(index: Int): UInt {\n
AbstractList.checkElementIndex(index, size)\n        return this@asList[index]\n    }\n    override fun
indexOf(element: UInt): Int {\n        if ((element as Any?) !is UInt) return -1\n        return
this@asList.indexOf(element)\n    }\n    override fun lastIndexOf(element: UInt): Int {\n        if ((element as
Any?) !is UInt) return -1\n        return this@asList.lastIndexOf(element)\n    }\n}\n}\n\n/**\n * Returns a
[List] that wraps the original array.\n */\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic actual fun
ULongArray.asList(): List<ULong> {\n
return object : AbstractList<ULong>(), RandomAccess {\n    override

```


modifier instead", level = DeprecationLevel.ERROR)\r\npublic annotation class native(public val name: String = "\")\r\n\r\n@Target(FUNCTION)\r\n\r\n@Deprecated("Use inline extension function with body using dynamic")\r\npublic annotation class nativeGetter\r\n\r\n\r\n@Target(FUNCTION)\r\n\r\n@Deprecated("Use inline extension function with body using dynamic")\r\npublic annotation class nativeSetter\r\n\r\n\r\n@Target(FUNCTION)\r\n\r\n@Deprecated("Use inline extension function with body using dynamic")\r\npublic annotation class nativeInvoke\r\n\r\n\r\n@Target(CLASS, FUNCTION, PROPERTY)\r\n\r\ninternal annotation class library(public val name: String = "\")\r\n\r\n@Target(CLASS)\r\n\r\ninternal annotation class marker\r\n\r\n\r\n/**\r\n * Gives a declaration (a function, a property or a class) specific name in JavaScript.\r\n * \r\n * This may be useful in the following cases:\r\n * \r\n * * There are two functions for which the compiler gives same name in JavaScript, you can\r\n * mark one with `@JsName(...)` to prevent the compiler from reporting error.\r\n * \r\n * * You are writing a JavaScript library in Kotlin. The compiler produces mangled names\r\n * for functions with parameters, which is unnatural for usual JavaScript developer.\r\n * You can put `@JsName(...)` on functions you want to be available from JavaScript.\r\n * \r\n * * For some reason you want to rename declaration, e.g. there's common term in JavaScript\r\n * for a concept provided by the declaration, which is uncommon in Kotlin.\r\n * \r\n * Example:\r\n * \r\n * ``` kotlin\r\n * class Person(val name: String) {\r\n * fun hello() {\r\n * println("Hello \$name!")\r\n * }\r\n * \r\n * @JsName("helloWithGreeting")\r\n * fun hello(greeting: String) {\r\n * println("\$greeting \$name!")\r\n * }\r\n * }\r\n * ```\r\n * \r\n * @property name the name which compiler uses both for declaration itself and for all references to the declaration.\r\n * It's required to denote a valid JavaScript identifier.\r\n * \r\n * @Retention(AnnotationRetention.BINARY)\r\n * @Target(CLASS, FUNCTION, PROPERTY, CONSTRUCTOR, PROPERTY_GETTER, PROPERTY_SETTER)\r\n * public actual annotation class JsName(actual val name: String)\r\n\r\n\r\n/**\r\n * Denotes an `external` declaration that must be imported from native JavaScript library.\r\n * \r\n * The compiler produces the code relevant for the target module system, for example, in case of CommonJS,\r\n * it will import the declaration via the `require(...)` function.\r\n * \r\n * \r\n * The annotation can be used on top-level external declarations (classes, properties, functions) and files.\r\n * \r\n * In case of file (which can't be `external`) the following rule applies: all the declarations in\r\n * the file must be `external`. By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object\r\n * that contain all the declarations from the file.\r\n * \r\n * Example:\r\n * \r\n * ``` kotlin\r\n * @JsModule("jquery")\r\n * external abstract class JQuery() {\r\n * // some declarations here\r\n * }\r\n * \r\n * @JsModule("jquery")\r\n * external fun JQuery(element: Element): JQuery\r\n * ```\r\n * \r\n * \r\n * @property import name of a module to import declaration from.\r\n * It is not interpreted by the Kotlin compiler, it's passed as is directly to the target module system.\r\n * \r\n * @see JsNonModule\r\n * \r\n * @Retention(AnnotationRetention.BINARY)\r\n * @Target(CLASS, PROPERTY, FUNCTION, FILE)\r\n * public annotation class JsModule(val import: String)\r\n\r\n\r\n/**\r\n * Denotes an `external` declaration that can be used without module system.\r\n * \r\n * By default, an `external` declaration is available regardless your target module system.\r\n * \r\n * However, by applying [JsModule] annotation you can make a declaration unavailable to *plain* module system.\r\n * \r\n * Some JavaScript libraries are distributed both as a standalone downloadable piece of JavaScript and as a module available\r\n * as an npm package.\r\n * \r\n * To tell the Kotlin compiler to accept both cases, you can augment [JsModule] with the `@JsNonModule` annotation.\r\n * \r\n * For example:\r\n * \r\n * ``` kotlin\r\n * @JsModule("jquery")\r\n * @JsNonModule\r\n * @JsName("\$")\r\n * external abstract class JQuery() {\r\n * // some declarations here\r\n * }\r\n * \r\n * @JsModule("jquery")\r\n * @JsNonModule\r\n * @JsName("\$")\r\n * external fun JQuery(element: Element): JQuery\r\n * ```\r\n * \r\n * \r\n * @see JsModule\r\n * \r\n * @Retention(AnnotationRetention.BINARY)\r\n * @Target(CLASS, PROPERTY, FUNCTION, FILE)\r\n * public annotation class JsNonModule\r\n\r\n\r\n/**\r\n * Adds prefix to `external` declarations in a source file.\r\n * \r\n * JavaScript does not have concept of packages (namespaces). They are usually emulated by nested objects.\r\n * \r\n * The compiler turns references to `external` declarations either to plain unprefixed names (in case of *plain* modules)\r\n * or to plain imports.\r\n * \r\n * However, if a JavaScript library provides its declarations in packages, you won't be satisfied with this.\r\n * \r\n * You can tell the compiler to generate additional prefix before references to `external` declarations using the `@JsQualifier(...)`\r\n * annotation.\r\n * \r\n * Note that a file marked with the

`@JsQualifier(...)` annotation can't contain non-`external` declarations.
 Example:
`@file:JsQualifier("my.jsPackageName") package some.kotlinPackage external fun foo(x: Int) external fun bar(): String`
 The qualifier to add to the declarations in the generated code. It must be a sequence of valid JavaScript identifiers separated by the ``` character.
 Examples of valid qualifiers are: ``foo``, ``bar.Baz``, ``_.$0.f``
`@Retention(AnnotationRetention.BINARY) @Target(AnnotationTarget.FILE) public annotation class JsQualifier(val value: String)`
 Exports top-level declaration. Used in future IR-based backend. Has no effect in current JS backend.
`@Retention(AnnotationRetention.BINARY) @Target(CLASS, PROPERTY, FUNCTION, FILE) public annotation class JsExport("","")`
 Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
`package kotlin.jvm` these are used in common generated code in `stdlib`
 TODO: find how to deprecate these
`@Target(AnnotationTarget.FIELD) @Retention(AnnotationRetention.SOURCE) public actual annotation class Volatile @Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY_GETTER, AnnotationTarget.PROPERTY_SETTER) @Retention(AnnotationRetention.SOURCE) public actual annotation class Synchronized("","")`
 Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
`package kotlin.collections`
 Provides a skeletal implementation of the `[MutableCollection]` interface.
`@param E` the type of elements contained in the collection. The collection is invariant on its element type.
 public actual abstract class `AbstractMutableCollection<E>` protected actual constructor() : `AbstractCollection<E>()`, `MutableCollection<E>` {
 actual abstract override fun `add(element: E): Boolean`
 actual override fun `remove(element: E): Boolean` {
 val iterator = iterator()
 while (iterator.hasNext()) {
 if (iterator.next() == element) {
 iterator.remove()
 return true
 }
 }
 return false
 }
 actual override fun `addAll(elements: Collection<E>): Boolean` {
 var modified = false
 for (element in elements) {
 if (add(element)) modified = true
 }
 return modified
 }
 actual override fun `removeAll(elements: Collection<E>): Boolean` = (this as `MutableIterable<E>`).`removeAll` { it in elements }
 actual override fun `retainAll(elements: Collection<E>): Boolean` = (this as `MutableIterable<E>`).`removeAll` { it !in elements }
 actual override fun `clear(): Unit` {
 val iterator = this.iterator()
 while (iterator.hasNext()) {
 iterator.next()
 iterator.remove()
 }
 }
`@JsName("toJSON") open fun toJSON(): Any = this.toArray()`
 Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 Based on GWT
`AbstractList`
 Copyright 2007 Google Inc.
`package kotlin.collections`
 Provides a skeletal implementation of the `[MutableList]` interface.
`@param E` the type of elements contained in the list. The list is invariant on its element type.
 public actual abstract class `AbstractMutableList<E>` protected actual constructor() : `AbstractMutableCollection<E>()`, `MutableList<E>` {
 protected var `modCount: Int` = 0
 abstract override fun `add(index: Int, element: E): Unit`
 abstract override fun `removeAt(index: Int): E`
 abstract override fun `set(index: Int, element: E): E`
 /*
 * Adds the specified element to the end of this list.
 * @return `true` because the list is always modified as the result of this operation.
 */
 actual override fun `add(element: E): Boolean` {
 add(size, element)
 return true
 }
 actual override fun `addAll(index: Int, elements: Collection<E>): Boolean` {
 var `_index` = index
 var `changed` = false
 for (e in elements) {
 add(_index++, e)
 changed = true
 }
 return changed
 }
 actual override fun `clear()` {
 removeRange(0, size)
 }
 actual override fun `removeAll(elements: Collection<E>): Boolean` = `removeAll` { it in elements }
 actual override fun `retainAll(elements: Collection<E>): Boolean` = `removeAll` { it !in elements }
 actual override fun `iterator(): MutableIterator<E>` = `IteratorImpl()`
 actual override fun `contains(element: E): Boolean` =

```

indexOf(element) >= 0\r\n\r\n    actual override fun indexOf(element: E): Int {\r\n        for (index in 0..lastIndex)
{\r\n            if (get(index) == element) {\r\n                return index\r\n            }\r\n        }\r\n        return -1\r\n    }\r\n\r\n    actual override fun lastIndexOf(element: E): Int {\r\n        for (index in lastIndex downTo 0) {\r\n            if (get(index) == element) {\r\n                return index\r\n            }\r\n        }\r\n        return -1\r\n    }\r\n\r\n    actual
override fun listIterator(): MutableListIterator<E> = listIterator(0)\r\n    actual override fun listIterator(index: Int):
MutableListIterator<E> = ListIteratorImpl(index)\r\n\r\n    actual override fun subList(fromIndex: Int, toIndex:
Int): MutableList<E> = SubList(this, fromIndex, toIndex)\r\n\r\n    /**\r\n     * Removes the range of elements from
this list starting from [fromIndex] and ending with but not including [toIndex].\r\n     */\r\n    protected open fun
removeRange(fromIndex: Int, toIndex: Int) {\r\n        val iterator = listIterator(fromIndex)\r\n        repeat(toIndex -
fromIndex) {\r\n            iterator.next()\r\n            iterator.remove()\r\n        }\r\n    }\r\n\r\n    /**\r\n     * Compares
this list with another list instance with the ordered structural equality.\r\n     */\r\n     * @return true, if [other]
instance is a [List] of the same size, which contains the same elements in the same order.\r\n     */\r\n    override fun
equals(other: Any?): Boolean {\r\n        if (other === this) return true\r\n        if (other !is List<*>) return
false\r\n        return AbstractList.orderedEquals(this, other)\r\n    }\r\n\r\n    /**\r\n     * Returns the hash code
value for this list.\r\n     */\r\n    override fun hashCode(): Int = AbstractList.orderedHashCode(this)\r\n\r\n    private
open inner class IteratorImpl : MutableListIterator<E> {\r\n        /** the index of the item that will be returned
on the next call to [next]`() */\r\n        protected var index = 0\r\n        /** the index of the item that was returned
on the previous call to [next]`() * or [ListIterator.previous]`() (for `ListIterator`),\r\n        * -1 if no such item
exists\r\n        */\r\n        protected var last = -1\r\n\r\n        override fun hasNext(): Boolean = index < size\r\n\r\n
        override fun next(): E {\r\n            if (!hasNext()) throw NoSuchElementException()\r\n            last = index++\r\n
            return get(last)\r\n        }\r\n\r\n        override fun remove() {\r\n            check(last != -1) { "\"Call next() or
previous() before removing element from the iterator.\""}\r\n            removeAt(last)\r\n            index = last\r\n
            last = -1\r\n        }\r\n    }\r\n\r\n    /**\r\n     * Implementation of `MutableListIterator` for abstract lists.\r\n
    */\r\n    private inner class ListIteratorImpl(index: Int) : IteratorImpl(), MutableListIterator<E> {\r\n        init
{\r\n            AbstractList.checkPositionIndex(index, this@AbstractMutableList.size)\r\n            this.index = index\r\n
        }\r\n\r\n        override fun hasPrevious(): Boolean = index > 0\r\n\r\n        override fun nextIndex(): Int =
index\r\n\r\n        override fun previous(): E {\r\n            if (!hasPrevious()) throw NoSuchElementException()\r\n\r\n
            last = --index\r\n            return get(last)\r\n        }\r\n\r\n        override fun previousIndex(): Int = index -
1\r\n\r\n        override fun add(element: E) {\r\n            add(index, element)\r\n            index++\r\n            last = -1\r\n
        }\r\n\r\n        override fun set(element: E) {\r\n            check(last != -1) { "\"Call next() or previous() before
updating element value with the iterator.\""}\r\n            this@AbstractMutableList[last] = element\r\n        }\r\n    }\r\n\r\n
    private class SubList<E>(private val list: AbstractMutableList<E>, private val fromIndex: Int, toIndex:
Int) : AbstractMutableList<E>(), RandomAccess {\r\n        private var _size: Int = 0\r\n\r\n        init {\r\n
            AbstractList.checkRangeIndexes(fromIndex, toIndex, list.size)\r\n            this._size = toIndex - fromIndex\r\n
        }\r\n\r\n        override fun add(index: Int, element: E) {\r\n            AbstractList.checkPositionIndex(index,
_size)\r\n            list.add(fromIndex + index, element)\r\n            _size++\r\n        }\r\n\r\n        override fun
get(index: Int): E {\r\n            AbstractList.checkElementIndex(index, _size)\r\n            return list[fromIndex +
index]\r\n        }\r\n\r\n        override fun removeAt(index: Int): E {\r\n            AbstractList.checkElementIndex(index, _size)\r\n\r\n
            val result = list.removeAt(fromIndex + index)\r\n            _size--\r\n            return result\r\n        }\r\n\r\n
        override fun set(index: Int, element: E): E {\r\n            AbstractList.checkElementIndex(index, _size)\r\n\r\n
            return list.set(fromIndex + index, element)\r\n        }\r\n\r\n        override val size: Int get() = _size\r\n
    }\r\n\r\n}\r\n\r\n"/**\r\n * Copyright 2010-2018 JetBrains s.r.o.
and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license
that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n/**\r\n * Based on GWT AbstractMap\r\n * Copyright
2007 Google Inc.\r\n */\r\n\r\npackage kotlin.collections\r\n\r\n/**\r\n * Provides a skeletal implementation of the
[MutableMap] interface.\r\n */\r\n * The implementor is required to implement [entries] property, which should
return mutable set of map entries, and [put] function.\r\n */\r\n * @param K the type of map keys. The map is
invariant on its key type.\r\n * @param V the type of map values. The map is invariant on its value type.\r\n

```

```

*/\r\npublic actual abstract class AbstractMutableMap<K, V> protected actual constructor() : AbstractMap<K, V>(),
MutableMap<K, V> {\r\n\r\n    /**\r\n     * A mutable [Map.Entry] shared by several [Map] implementations.\r\n
*/\r\n    internal open class SimpleEntry<K, V>(override val key: K, value: V) : MutableMap.MutableEntry<K, V>
{\r\n        constructor(entry: Map.Entry<K, V>) : this(entry.key, entry.value)\r\n\r\n        private var _value =
value\r\n\r\n        override val value: V get() = _value\r\n\r\n        override fun setValue(newValue: V): V {\r\n
val oldValue = this._value\r\n            this._value = newValue\r\n            return oldValue\r\n        }\r\n\r\n
override fun hashCode(): Int = entryHashCode(this)\r\n        override fun toString(): String = entryToString(this)\r\n
        override fun equals(other: Any?): Boolean = entryEquals(this, other)\r\n\r\n    }\r\n\r\n    actual override fun
clear() {\r\n        entries.clear()\r\n    }\r\n\r\n    private var _keys: MutableSet<K>? = null\r\n    actual override val
keys: MutableSet<K>\r\n        get() {\r\n            if (_keys == null) {\r\n                _keys = object :
AbstractMutableSet<K>() {\r\n                    override fun add(element: K): Boolean = throw
UnsupportedOperationException("Add is not supported on keys")\r\n                }\r\n            }\r\n
            override fun clear() {\r\n                this@AbstractMutableMap.clear()\r\n            }\r\n\r\n
            override operator fun contains(element:
K): Boolean = containsKey(element)\r\n\r\n            override operator fun iterator(): MutableIterator<K> {\r\n
                val entryIterator = entries.iterator()\r\n                return object : MutableIterator<K> {\r\n
                    override fun hasNext(): Boolean = entryIterator.hasNext()\r\n                    override fun next(): K =
entryIterator.next().key\r\n                    override fun remove() = entryIterator.remove()\r\n                }\r\n
            }\r\n\r\n            override fun remove(element: K): Boolean {\r\n                if
(containsKey(element)) {\r\n                    this@AbstractMutableMap.remove(element)\r\n
                    return true\r\n                }\r\n                return false\r\n            }\r\n\r\n
            override val size:
Int get() = this@AbstractMutableMap.size\r\n        }\r\n\r\n        return _keys!!\r\n    }\r\n\r\n
actual abstract override fun put(key: K, value: V): V?\r\n\r\n    actual override fun putAll(from: Map<out K, V>)
{\r\n        for ((key, value) in from) {\r\n            put(key, value)\r\n        }\r\n    }\r\n\r\n    private var _values:
MutableCollection<V>? = null\r\n    actual override val values: MutableCollection<V>\r\n        get() {\r\n            if
(_values == null) {\r\n                _values = object : AbstractMutableCollection<V>() {\r\n                    override fun
add(element: V): Boolean = throw UnsupportedOperationException("Add is not supported on values")\r\n\r\n
                    override fun clear() = this@AbstractMutableMap.clear()\r\n\r\n                    override operator fun
contains(element: V): Boolean = containsValue(element)\r\n\r\n                    override operator fun iterator():
MutableIterator<V> {\r\n                        val entryIterator = entries.iterator()\r\n                        return object :
MutableIterator<V> {\r\n                            override fun hasNext(): Boolean = entryIterator.hasNext()\r\n
                            override fun next(): V = entryIterator.next().value\r\n                            override fun remove() =
entryIterator.remove()\r\n                        }\r\n                    }\r\n\r\n                    override val size: Int get() =
this@AbstractMutableMap.size\r\n                }\r\n\r\n                // TODO: should we implement them this way? Currently it's
unspecified in JVM\r\n            }\r\n\r\n            override fun equals(other: Any?): Boolean {\r\n                if (this === other)
return true\r\n                if (other !is Collection<*>) return false\r\n                return
AbstractList.orderedEquals(this, other)\r\n            }\r\n\r\n            override fun hashCode(): Int =
AbstractList.orderedHashCode(this)\r\n        }\r\n\r\n        return _values!!\r\n    }\r\n\r\n    actual
override fun remove(key: K): V? {\r\n        val iter = entries.iterator()\r\n        while (iter.hasNext()) {\r\n            val
entry = iter.next()\r\n            val k = entry.key\r\n            if (key == k) {\r\n                val value = entry.value\r\n
                iter.remove()\r\n                return value\r\n            }\r\n        }\r\n        return null\r\n    }\r\n\r\n
}

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code
is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\npackage
kotlin.collections\r\n\r\n/**\r\n * Provides a skeletal implementation of the [MutableSet] interface.\r\n */\r\n *
@param E the type of elements contained in the set. The set is invariant on its element type.\r\n */\r\npublic actual
abstract class AbstractMutableSet<E> protected actual constructor() : AbstractMutableCollection<E>(),
MutableSet<E> {\r\n\r\n    /**\r\n     * Compares this set with another set instance with the unordered structural
equality.\r\n     */\r\n     * @return `true`, if [other] instance is a [Set] of the same size, all elements of which are
contained in this set.\r\n     */\r\n    override fun equals(other: Any?): Boolean {\r\n        if (other === this) return

```

```

true\r\n    if (other !is Set<*>) return false\r\n    return AbstractSet.setEquals(this, other)\r\n  }\r\n\r\n  /**\r\n   * Returns the hash code value for this set.\r\n   */\r\n  override fun hashCode(): Int =
AbstractSet.unorderedHashCode(this)\r\n\r\n}", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.collections\r\n\r\n/**\r\n * Provides a
[MutableList] implementation, which uses a resizable array as its backing storage.\r\n */\r\n * This implementation
doesn't provide a way to manage capacity, as backing JS array is resizeable itself.\r\n * There is no speed advantage
to pre-allocating array sizes in JavaScript, so this implementation does not include any of the\r\n * capacity and
\"growth increment\" concepts.\r\n */\r\n\r\npublic actual open class ArrayList<E> internal constructor(private var
array: Array<Any?>) : AbstractMutableList<E>(), MutableList<E>, RandomAccess {\r\n\r\n  /**\r\n   * Creates
an empty [ArrayList].\r\n   */\r\n  public actual constructor() : this(emptyArray()) {\r\n\r\n  /**\r\n   * Creates
an empty [ArrayList].\r\n   * @param initialCapacity initial capacity (ignored)\r\n   */\r\n  @Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")\r\n  public actual
constructor(initialCapacity: Int = 0) : this(emptyArray()) {\r\n\r\n  /**\r\n   * Creates an [ArrayList] filled from
the [elements] collection.\r\n   */\r\n  public actual constructor(elements: Collection<E>) :
this(elements.toArray<Any?>()) {\r\n\r\n  /** Does nothing in this ArrayList implementation. */\r\n  public actual
fun trimToSize() {\r\n\r\n  /** Does nothing in this ArrayList implementation. */\r\n  public actual
fun ensureCapacity(minCapacity: Int) {\r\n\r\n  actual override val size: Int get() = array.size\r\n  @Suppress(\"UNCHECKED_CAST\")\r\n  actual override fun get(index: Int): E = array[rangeCheck(index)] as
E\r\n  actual override fun set(index: Int, element: E): E {\r\n    rangeCheck(index)\r\n  @Suppress(\"UNCHECKED_CAST\")\r\n    return array[index].apply { array[index] = element } as E\r\n  }\r\n\r\n  actual override fun add(element: E): Boolean {\r\n    array.asDynamic().push(element)\r\n    modCount++\r\n    return true\r\n  }\r\n\r\n  actual override fun add(index: Int, element: E): Unit {\r\n    array.asDynamic().splice(insertionRangeCheck(index), 0, element)\r\n    modCount++\r\n  }\r\n\r\n  actual
override fun addAll(elements: Collection<E>): Boolean {\r\n    if (elements.isEmpty()) return false\r\n\r\n    array += elements.toArray<Any?>()\r\n    modCount++\r\n    return true\r\n  }\r\n\r\n  actual override
fun addAll(index: Int, elements: Collection<E>): Boolean {\r\n    insertionRangeCheck(index)\r\n\r\n    if
(index == size) return addAll(elements)\r\n    if (elements.isEmpty()) return false\r\n    when (index) {\r\n    size -> return addAll(elements)\r\n    0 -> array = elements.toArray<Any?>() + array\r\n    else ->
array = array.copyOfRange(0, index).asDynamic().concat(elements.toArray<Any?>(),
array.copyOfRange(index, size))\r\n  }\r\n\r\n  modCount++\r\n  return true\r\n  }\r\n\r\n  actual
override fun removeAt(index: Int): E {\r\n    rangeCheck(index)\r\n    modCount++\r\n    return if (index ==
lastIndex)\r\n      array.asDynamic().pop()\r\n    else\r\n      array.asDynamic().splice(index, 1)[0]\r\n  }\r\n\r\n  actual override fun remove(element: E): Boolean {\r\n    for (index in array.indices) {\r\n    if
(array[index] == element) {\r\n      array.asDynamic().splice(index, 1)\r\n      modCount++\r\n      return true\r\n    }\r\n  }\r\n  return false\r\n  }\r\n\r\n  override fun removeRange(fromIndex: Int,
toIndex: Int) {\r\n    modCount++\r\n    array.asDynamic().splice(fromIndex, toIndex - fromIndex)\r\n  }\r\n\r\n  actual override fun clear() {\r\n    array = emptyArray()\r\n    modCount++\r\n  }\r\n\r\n\r\n  actual
override fun indexOf(element: E): Int = array.indexOf(element)\r\n\r\n  actual override fun lastIndexOf(element:
E): Int = array.lastIndexOf(element)\r\n\r\n  override fun toString() = arrayToString(array)\r\n  override fun
toArray(): Array<Any?> = js(\"[]\").slice.call(array)\r\n\r\n\r\n\r\n  private fun rangeCheck(index: Int) = index.apply
{\r\n    AbstractList.checkElementIndex(index, size)\r\n  }\r\n\r\n  private fun insertionRangeCheck(index: Int)
= index.apply {\r\n    AbstractList.checkPositionIndex(index, size)\r\n  }\r\n}\r\n", "/*\r\n * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n\r\npackage
kotlin.collections\r\n\r\n\r\n@UseExperimental(ExperimentalUnsignedTypes::class)\r\n@SinceKotlin(\"1.3\")\r\n@
kotlin.js.JsName(\"contentDeepHashCodeImpl\")\r\n\r\ninternal fun <T> Array<out T>.contentDeepHashCodeImpl():
Int {\r\n  var result = 1\r\n  for (element in this) {\r\n    val elementHash = when {\r\n      element == null ->

```

```

0\r\n        isArrayish(element) -> (element.unsafeCast<Array<*>>()).contentDeepHashCodeImpl()\r\n\r\n
element is UByteArray -> element.contentHashCode()\r\n        element is UShortArray ->
element.contentHashCode()\r\n        element is UIntArray -> element.contentHashCode()\r\n        element is
ULongArray -> element.contentHashCode()\r\n\r\n        else -> element.hashCode()\r\n    }\r\n\r\n
    result = 31 * result + elementHash\r\n    }\r\n    return result\r\n}"/*\r\n * Copyright 2010-2019 JetBrains s.r.o.
and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license
that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.collections\r\n\r\ninternal fun <T>
sortArrayWith(array: Array<out T>, comparison: (T, T) -> Int) {\r\n    if (getStableSortingIsSupported()) {\r\n
array.asDynamic().sort(comparison)\r\n    } else {\r\n        mergeSort(array.unsafeCast<Array<T>>(), 0,
array.lastIndex, Comparator(comparison))\r\n    }\r\n}\r\n\r\ninternal fun <T> sortArrayWith(array: Array<out T>,
comparator: Comparator<in T>) {\r\n    if (getStableSortingIsSupported()) {\r\n        val comparison = { a: T, b: T ->
comparator.compare(a, b) }\r\n        array.asDynamic().sort(comparison)\r\n    } else {\r\n
mergeSort(array.unsafeCast<Array<T>>(), 0, array.lastIndex, comparator)\r\n    }\r\n}\r\n\r\ninternal fun <T :
Comparable<T>> sortArray(array: Array<out T>) {\r\n    if (getStableSortingIsSupported()) {\r\n        val
comparison = { a: T, b: T -> a.compareTo(b) }\r\n        array.asDynamic().sort(comparison)\r\n    } else {\r\n
mergeSort(array.unsafeCast<Array<T>>(), 0, array.lastIndex, naturalOrder())\r\n    }\r\n}\r\n\r\nprivate var
_stableSortingIsSupported: Boolean? = null\r\nprivate fun getStableSortingIsSupported(): Boolean {\r\n
_stableSortingIsSupported?.let { return it }\r\n    _stableSortingIsSupported = false\r\n\r\n    val array =
js("\r\n[]").unsafeCast<Array<Int>>()\r\n    // known implementations may use stable sort for arrays of up to 512
elements\r\n    // so we create slightly more elements to test stability\r\n    for (index in 0 until 600)
array.asDynamic().push(index)\r\n    val comparison = { a: Int, b: Int -> (a and 3) - (b and 3) }\r\n
array.asDynamic().sort(comparison)\r\n    for (index in 1 until array.size) {\r\n        val a = array[index - 1]\r\n
val b = array[index]\r\n        if ((a and 3) == (b and 3) && a >= b) return false\r\n    }\r\n
_stableSortingIsSupported = true\r\n    return true\r\n}\r\n\r\nprivate fun <T> mergeSort(array: Array<T>, start:
Int, endInclusive: Int, comparator: Comparator<in T>) {\r\n    val buffer =
arrayOfNulls<Any?>(array.size).unsafeCast<Array<T>>()\r\n    val result = mergeSort(array, buffer, start,
endInclusive, comparator)\r\n    if (result != array) {\r\n        result.forEachIndexed { i, v -> array[i] = v }\r\n
}\r\n}\r\n\r\n// Both start and end are inclusive indices.\r\nprivate fun <T> mergeSort(array: Array<T>, buffer:
Array<T>, start: Int, end: Int, comparator: Comparator<in T>): Array<T> {\r\n    if (start == end) {\r\n
return array\r\n    }\r\n\r\n    val median = (start + end) / 2\r\n    val left = mergeSort(array, buffer, start, median,
comparator)\r\n    val right = mergeSort(array, buffer, median + 1, end, comparator)\r\n\r\n    val target = if (left ==
buffer) array else buffer\r\n\r\n    // Merge.\r\n    var leftIndex = start\r\n    var rightIndex = median + 1\r\n    for (i in
start..end) {\r\n        when {\r\n            leftIndex <= median && rightIndex <= end -> {\r\n                val leftValue =
left[leftIndex]\r\n                val rightValue = right[rightIndex]\r\n\r\n                if (comparator.compare(leftValue,
rightValue) <= 0) {\r\n                    target[i] = leftValue\r\n                    leftIndex++\r\n                } else {\r\n
target[i] = rightValue\r\n                    rightIndex++\r\n                }\r\n            }\r\n            leftIndex <= median ->
{\r\n                target[i] = left[leftIndex]\r\n                leftIndex++\r\n            }\r\n            else /* rightIndex <= end */
-> {\r\n                target[i] = right[rightIndex]\r\n                rightIndex++\r\n                Unit // TODO: Fix KT-
31506\r\n            }\r\n        }\r\n        }\r\n    }\r\n    return target\r\n}"/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and
Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that
can be found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.collections\r\n\r\ninternal interface
EqualityComparator {\r\n    /**\r\n     * Subclasses must override to return a value indicating\r\n     * whether or not
two keys or values are equal.\r\n     */\r\n    abstract fun equals(value1: Any?, value2: Any?): Boolean\r\n\r\n
/**\r\n     * Subclasses must override to return the hash code of a given key.\r\n     */\r\n    abstract fun
getHashCode(value: Any?): Int\r\n\r\n    object HashCode : EqualityComparator {\r\n        override fun
equals(value1: Any?, value2: Any?): Boolean = value1 == value2\r\n\r\n        override fun getHashCode(value:
Any?): Int = value?.hashCode() ?: 0\r\n    }\r\n}"/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be

```

```

found in the license/LICENSE.txt file.
 * Based on GWT AbstractHashMap
 * Copyright 2008
Google Inc.
 kotlin.collections
 kotlin.collections.MutableMap.MutableEntry
 * Hash table based implementation of the [MutableMap]
interface.
 * This implementation makes no guarantees regarding the order of enumeration of [keys],
 [values] and [entries] collections.
 public actual open class HashMap<K, V> : AbstractMutableMap<K, V>,
 MutableMap<K, V> {
 private inner class EntrySet : AbstractMutableSet<MutableEntry<K, V>>() {
 override fun add(element: MutableEntry<K, V>): Boolean = throw UnsupportedOperationException("Add is not
supported on entries")
 override fun clear() {
 this@HashMap.clear()
 }
 override
operator fun contains(element: MutableEntry<K, V>): Boolean = containsEntry(element)
 override
operator fun iterator(): MutableIterator<MutableEntry<K, V>> = internalMap.iterator()
 override fun
remove(element: MutableEntry<K, V>): Boolean {
 if (contains(element)) {
 this@HashMap.remove(element.key)
 return true
 }
 return false
 }
 override val size: Int get() = this@HashMap.size
 }
 /**
 * Internal implementation of the map:
either string-based or hashcode-based.
 */
 private val internalMap: InternalMap<K, V>
 private val equality: EqualityComparator
 internal constructor(internalMap: InternalMap<K, V>) : super() {
 this.internalMap = internalMap
 this.equality = internalMap.equality
 }
 /**
 * Constructs
an empty [HashMap] instance.
 */
 actual constructor() :
this(InternalHashMap(EqualityComparator.HashCode))
 /**
 * Constructs an empty [HashMap]
instance.
 * @param initialCapacity the initial capacity (ignored)
 * @param loadFactor the
load factor (ignored)
 * @throws IllegalArgumentException if the initial capacity or load factor are
negative
 * @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
 actual
constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : this() // This implementation of HashMap has
no need of load factors or capacities.
 require(initialCapacity >= 0) { "Negative initial capacity:
$initialCapacity" }
 require(loadFactor >= 0) { "Non-positive load factor: $loadFactor" }
 }
 actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)
 /**
 * Constructs an instance of
[HashMap] filled with the contents of the specified [original] map.
 */
 actual constructor(original:
Map<out K, V>) : this() {
 this.putAll(original)
 }
 actual override fun clear() {
 internalMap.clear()
 structureChanged(this)
 }
 actual override fun containsKey(key: K):
Boolean = internalMap.contains(key)
 actual override fun containsValue(value: V): Boolean =
internalMap.any { equality.equals(it.value, value) }
 private var _entries:
MutableSet<MutableMap.MutableEntry<K, V>>? = null
 actual override val entries:
MutableSet<MutableMap.MutableEntry<K, V>>
 get() {
 if (_entries == null) {
 _entries = createEntrySet()
 }
 return _entries!!
 }
 protected open fun
createEntrySet(): MutableSet<MutableMap.MutableEntry<K, V>> = EntrySet()
 actual override operator
fun get(key: K): V? = internalMap.get(key)
 actual override fun put(key: K, value: V): V? =
internalMap.put(key, value)
 actual override fun remove(key: K): V? = internalMap.remove(key)
 actual override val size: Int get() = internalMap.size
 }
 /**
 * Constructs the specialized
implementation of [HashMap] with [String] keys, which stores the keys as properties of
 * JS object without
hashing them.
 */
 public fun <V> stringMapOf(vararg pairs: Pair<String, V>): HashMap<String, V> {
 return HashMap<String, V>(InternalStringMap(EqualityComparator.HashCode)).apply { putAll(pairs)
 }
 }
 /**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of
this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 *
 * Based on GWT HashSet
 * Copyright 2008 Google Inc.
 kotlin.collections
 * The implementation of the [MutableSet] interface, backed by a [HashMap]
instance.
 */
 public actual open class HashSet<E> : AbstractMutableSet<E>, MutableSet<E> {
 private val map: HashMap<E, Any>
 /**
 * Constructs a new empty [HashSet].
 */
 actual
constructor() {
 map = HashMap<E, Any>()
 }
 /**
 * Constructs a new [HashSet] filled
with the elements of the specified collection.
 */
 actual constructor(elements: Collection<E>) {

```



```

map = HashMap<E, Any>(elements.size)\r\n    addAll(elements)\r\n } \r\n\r\n /**\r\n  * Constructs a new
empty [HashSet].\r\n  * @param initialCapacity the initial capacity (ignored)\r\n  * @param loadFactor
the load factor (ignored)\r\n  * @throws IllegalArgumentException if the initial capacity or load factor
are negative\r\n  */\r\n @SuppressWarnings("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\n actual
constructor(initialCapacity: Int, loadFactor: Float = 0.0f) {\r\n    map = HashMap<E, Any>(initialCapacity,
loadFactor)\r\n } \r\n\r\n actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)\r\n\r\n /**\r\n  *
Protected constructor to specify the underlying map. This is used by\r\n  * LinkedHashMap.\r\n  * @param
map underlying map to use.\r\n  */\r\n internal constructor(map: HashMap<E, Any>) {\r\n    this.map =
map\r\n } \r\n\r\n actual override fun add(element: E): Boolean {\r\n    val old = map.put(element, this)\r\n
return old == null\r\n } \r\n\r\n actual override fun clear() {\r\n    map.clear()\r\n } \r\n\r\n// public override
fun clone(): Any {\r\n//    return HashSet<E>(this)\r\n// } \r\n\r\n actual override operator fun
contains(element: E): Boolean = map.containsKey(element)\r\n\r\n actual override fun isEmpty(): Boolean =
map.isEmpty()\r\n\r\n actual override fun iterator(): MutableIterator<E> = map.keys.iterator()\r\n\r\n actual
override fun remove(element: E): Boolean = map.remove(element) != null\r\n\r\n actual override val size: Int get()
= map.size\r\n\r\n}\r\n\r\n/**\r\n  * Creates a new instance of the specialized implementation of [HashSet] with the
specified [String] elements,\r\n  * which elements the keys as properties of JS object without hashing them.\r\n
*/\r\n\r\npublic fun stringSetOf(vararg elements: String): HashSet<String> {\r\n    return
HashSet(stringMapOf<Any>()).apply { addAll(elements) } \r\n}\r\n\r\n"/\r\n\r\n * Copyright 2010-2018 JetBrains s.r.o.
and Kotlin Programming Language contributors.\r\n  * Use of this source code is governed by the Apache 2.0 license
that can be found in the license/LICENSE.txt file.\r\n  */\r\n\r\n * Based on GWT InternalHashCodeMap\r\n  *
Copyright 2008 Google Inc.\r\n  */\r\n\r\n\r\npackage kotlin.collections\r\n\r\nimport
kotlin.collections.MutableMap.MutableEntry\r\n\r\nimport
kotlin.collections.AbstractMutableMap.SimpleEntry\r\n\r\n\r\n/**\r\n  * A simple wrapper around JavaScriptObject to
provide [java.util.Map]-like semantics for any\r\n  * key type.\r\n  * @param K key type.\r\n  * @param V value type.\r\n
*/\r\n\r\n * Implementation notes:\r\n  * @param K key type.\r\n  * @param V value type.\r\n  * A key's hashCode is the index in backingMap which should contain that key. Since several keys may\r\n  * have the
same hash, each value in hashCodeMap is actually an array containing all entries whose\r\n  * keys share the same
hash.\r\n  */\r\n\r\ninternal class InternalHashCodeMap<K, V>(override val equality: EqualityComparator) :
InternalMap<K, V> {\r\n\r\n    private var backingMap: dynamic = createJsMap()\r\n    override var size: Int = 0\r\n\r\n    private set\r\n\r\n    override fun put(key: K, value: V): V? {\r\n        val hashCode =
equality.getHashCode(key)\r\n        val chainOrEntry = getChainOrEntryOrNull(hashCode)\r\n        if (chainOrEntry
== null) {\r\n            // This is a new chain, put it to the map.\r\n            backingMap[hashCode] = SimpleEntry(key,
value)\r\n        } else {\r\n            if (chainOrEntry !is Array<*>) {\r\n                // It is an entry\r\n                val
entry: SimpleEntry<K, V> = chainOrEntry\r\n                if (equality.equals(entry.key, key)) {\r\n                    return
entry.setValue(value)\r\n                } else {\r\n                    backingMap[hashCode] = arrayOf(entry,
SimpleEntry(key, value))\r\n                    size++\r\n                    return null\r\n                } \r\n            } else {\r\n
                // Chain already exists, perhaps key also exists.\r\n                val chain: Array<MutableEntry<K, V>> =
chainOrEntry\r\n                val entry = chain.findEntryInChain(key)\r\n                if (entry != null) {\r\n
                    return entry.setValue(value)\r\n                } \r\n                chain.asDynamic().push(SimpleEntry(key, value))\r\n
            } \r\n        } \r\n        size++\r\n        // structureChanged(host)\r\n        return null\r\n    } \r\n\r\n    override fun
remove(key: K): V? {\r\n        val hashCode = equality.getHashCode(key)\r\n        val chainOrEntry =
getChainOrEntryOrNull(hashCode) ?: return null\r\n        if (chainOrEntry !is Array<*>) {\r\n            val entry:
MutableEntry<K, V> = chainOrEntry\r\n            if (equality.equals(entry.key, key)) {\r\n
                jsDeleteProperty(backingMap, hashCode)\r\n                size--\r\n                return entry.value\r\n            } else {\r\n
                return null\r\n            } \r\n        } else {\r\n            val chain: Array<MutableEntry<K, V>> = chainOrEntry\r\n
            for (index in chain.indices) {\r\n                val entry = chain[index]\r\n                if (equality.equals(key,
entry.key)) {\r\n                    if (chain.size == 1) {\r\n                        chain.asDynamic().length = 0\r\n
                    } \r\n                    // remove the whole array\r\n                    jsDeleteProperty(backingMap, hashCode)\r\n                } else {\r\n
                    // splice out the entry we're removing\r\n                    chain.asDynamic().splice(index, 1)\r\n
                } \r\n            } \r\n        } \r\n    } \r\n\r\n}

```



```

correct.\r\n// * This is to optimize for the common scenario where the values are not modified during\r\n// *
iterations where the entries are never stale.\r\n// */\r\n// private var valueMod: Int = 0\r\n\r\n override operator
fun contains(key: K): Boolean {\r\n    if (key !is String) return false\r\n    return backingMap[key] !==
undefined\r\n }\r\n\r\n override operator fun get(key: K): V? {\r\n    if (key !is String) return null\r\n    val
value = backingMap[key]\r\n    return if (value !== undefined) value.unsafeCast<V>() else null\r\n }\r\n\r\n\r\n
override fun put(key: K, value: V): V? {\r\n    require(key is String)\r\n    val oldValue = backingMap[key]\r\n
    backingMap[key] = value\r\n\r\n    if (oldValue === undefined) {\r\n        size++\r\n//
structureChanged(host)\r\n        return null\r\n    } else {\r\n//        valueMod++\r\n        return
oldValue.unsafeCast<V>()\r\n    }\r\n }\r\n\r\n override fun remove(key: K): V? {\r\n    if (key !is String)
return null\r\n    val value = backingMap[key]\r\n    if (value !== undefined) {\r\n
jsDeleteProperty(backingMap, key)\r\n        size--\r\n//        structureChanged(host)\r\n        return
value.unsafeCast<V>()\r\n    } else {\r\n//        valueMod++\r\n        return null\r\n    }\r\n }\r\n\r\n\r\n
override fun clear() {\r\n    backingMap = createJsMap()\r\n    size = 0\r\n }\r\n\r\n\r\n override fun
iterator(): MutableIterator<MutableEntry<K, V>> {\r\n    return object : MutableIterator<MutableEntry<K, V>>
{\r\n        private val keys: Array<String> = js("Object").keys(backingMap)\r\n        private val iterator =
keys.iterator()\r\n        private var lastKey: String? = null\r\n\r\n        override fun hasNext(): Boolean =
iterator.hasNext()\r\n\r\n        override fun next(): MutableEntry<K, V> {\r\n            val key = iterator.next()\r\n
            lastKey = key\r\n            @Suppress("UNCHECKED_CAST")\r\n            return newMapEntry(key
as K)\r\n        }\r\n\r\n        override fun remove() {\r\n            @Suppress("UNCHECKED_CAST")\r\n
            this@InternalStringMap.remove(checkNotNull(lastKey) as K)\r\n        }\r\n    }\r\n }\r\n\r\n private fun
newMapEntry(key: K): MutableEntry<K, V> = object : MutableEntry<K, V> {\r\n    override val key: K get() =
key\r\n    override val value: V get() = this@InternalStringMap[key].unsafeCast<V>()\r\n\r\n    override fun
setValue(newValue: V): V = this@InternalStringMap.put(key, newValue).unsafeCast<V>()\r\n\r\n    override fun
hashCode(): Int = AbstractMap.entryHashCode(this)\r\n    override fun toString(): String =
AbstractMap.entryToString(this)\r\n    override fun equals(other: Any?): Boolean =
AbstractMap.entryEquals(this, other)\r\n }\r\n}\r\n"/>\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n */\r\n*/\r\n * Based on GWT LinkedHashMap\r\n * Copyright 2008
Google Inc.\r\n */\r\npackage kotlin.collections\r\n\r\nimport
kotlin.collections.MutableMap.MutableEntry\r\n\r\n/**\r\n * Hash table based implementation of the [MutableMap]
interface, which additionally preserves the insertion order\r\n * of entries during the iteration.\r\n * The
insertion order is preserved by maintaining a doubly-linked list of all of its entries.\r\n */\r\npublic actual open class
LinkedHashMap<K, V> : HashMap<K, V>, MutableMap<K, V> {\r\n    /**\r\n     * The entry we use includes
next/prev pointers for a doubly-linked circular\r\n     * list with a head node. This reduces the special cases we have
to deal with\r\n     * in the list operations.\r\n     * Note that we duplicate the key from the underlying hash map
so we can find\r\n     * the eldest entry. The alternative would have been to modify HashMap so more\r\n     * of the
code was directly usable here, but this would have added some\r\n     * overhead to HashMap, or to reimplement
most of the HashMap code here with\r\n     * small modifications. Paying a small storage cost only if you use\r\n
     * LinkedHashMap and minimizing code size seemed like a better tradeoff\r\n     */\r\n    private class ChainEntry<K,
V>(key: K, value: V) : AbstractMutableMap.SimpleEntry<K, V>(key, value) {\r\n        internal var next:
ChainEntry<K, V>? = null\r\n        internal var prev: ChainEntry<K, V>? = null\r\n    }\r\n\r\n    private inner class
EntrySet : AbstractMutableSet<MutableEntry<K, V>>() {\r\n        private inner class EntryIterator :
MutableIterator<MutableEntry<K, V>> {\r\n            // The last entry that was returned from this iterator.\r\n\r\n
            private var last: ChainEntry<K, V>? = null\r\n\r\n            // The next entry to return from this iterator.\r\n\r\n
            private var next: ChainEntry<K, V>? = null\r\n\r\n            init {\r\n                next = head\r\n//
            recordLastKnownStructure(map, this)\r\n            }\r\n\r\n            override fun hasNext(): Boolean {\r\n
                return next !== null\r\n            }\r\n\r\n            override fun next(): MutableEntry<K, V> {\r\n//
                checkStructuralChange(map, this)\r\n                if (!hasNext()) throw NoSuchElementException()\r\n\r\n

```

```

val current = next!!\r\n          last = current\r\n          next = current.next.takeIf { it !== head }\r\n
return current\r\n          }\r\n\r\n          override fun remove() {\r\n          check(last != null)\r\n//
checkStructuralChange(map, this)\r\n\r\n          last!!.remove()\r\n          map.remove(last!!.key)\r\n//
recordLastKnownStructure(map, this)\r\n          last = null\r\n          }\r\n          }\r\n\r\n          override fun
add(element: MutableEntry<K, V>): Boolean = throw UnsupportedOperationException("Add is not supported on
entries")\r\n          override fun clear() {\r\n          this@LinkedHashMap.clear()\r\n          }\r\n\r\n          override
operator fun contains(element: MutableEntry<K, V>): Boolean = containsEntry(element)\r\n\r\n          override
operator fun iterator(): MutableIterator<MutableEntry<K, V>> = EntryIterator()\r\n\r\n          override fun
remove(element: MutableEntry<K, V>): Boolean {\r\n          if (contains(element)) {\r\n
this@LinkedHashMap.remove(element.key)\r\n          return true\r\n          }\r\n          return false\r\n
}\r\n\r\n          override val size: Int get() = this@LinkedHashMap.size\r\n          }\r\n\r\n\r\n          /*\r\n          * The head of the
insert order chain, which is a doubly-linked circular\r\n          * list.\r\n          * \r\n          * The most recently inserted node is at the
end of the chain, ie.\r\n          * chain.prev.\r\n          * \r\n          * private var head: ChainEntry<K, V>? = null\r\n\r\n          /**\r\n          *
Add this node to the end of the chain.\r\n          * \r\n          * private fun ChainEntry<K, V>.addToEnd() {\r\n          // This
entry is not in the list.\r\n          check(next == null && prev == null)\r\n\r\n          val _head = head\r\n          if (_head
== null) {\r\n          head = this\r\n          next = this\r\n          prev = this\r\n          } else {\r\n          // Chain is
valid.\r\n          val _tail = checkNotNull(_head.prev)\r\n          // Update me.\r\n          prev = _tail\r\n          next
= _head\r\n          // Update my new siblings: current head and old tail\r\n          _head.prev = this\r\n
_tail.next = this\r\n          }\r\n          }\r\n\r\n          /**\r\n          * Remove this node from the chain it is a part of.\r\n          * \r\n
private fun ChainEntry<K, V>.remove() {\r\n          if (this.next === this) {\r\n          // if this is single element,
remove head\r\n          head = null\r\n          } else {\r\n          if (head === this) {\r\n          // if this is first
element, move head to next\r\n          head = next\r\n          }\r\n          next!!.prev = prev\r\n          prev!!.next
= next\r\n          }\r\n          next = null\r\n          prev = null\r\n          }\r\n\r\n          /*\r\n          * The hashmap that keeps track of our
entries and the chain. Note that we\r\n          * duplicate the key here to eliminate changes to HashMap and minimize
the\r\n          * code here, at the expense of additional space.\r\n          * \r\n          private val map: HashMap<K, ChainEntry<K,
V>>\r\n\r\n          /**\r\n          * Constructs an empty [LinkedHashMap] instance.\r\n          * \r\n          actual constructor() :
super() {\r\n          map = HashMap<K, ChainEntry<K, V>>()\r\n          }\r\n\r\n          internal constructor(backingMap:
HashMap<K, Any>) : super() {\r\n          @SuppressWarnings("UNCHECKED_CAST") // expected to work due to
erasure\r\n          map = backingMap as HashMap<K, ChainEntry<K, V>>\r\n          }\r\n\r\n          /**\r\n          * Constructs an
empty [LinkedHashMap] instance.\r\n          * \r\n          * @param initialCapacity the initial capacity (ignored)\r\n          *
@param loadFactor the load factor (ignored)\r\n          * \r\n          * @throws IllegalArgumentException if the initial
capacity or load factor are negative\r\n          * \r\n
@SuppressWarnings("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\n          actual
constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : super(initialCapacity, loadFactor) {\r\n          map =
HashMap<K, ChainEntry<K, V>>()\r\n          }\r\n\r\n          actual constructor(initialCapacity: Int) : this(initialCapacity,
0.0f)\r\n\r\n          /**\r\n          * Constructs an instance of [LinkedHashMap] filled with the contents of the specified
[original] map.\r\n          * \r\n          actual constructor(original: Map<out K, V>) {\r\n          map = HashMap<K,
ChainEntry<K, V>>()\r\n          this.putAll(original)\r\n          }\r\n\r\n          actual override fun clear() {\r\n
map.clear()\r\n          head = null\r\n          }\r\n\r\n\r\n          override fun clone(): Any {\r\n//          return
LinkedHashMap(this)\r\n//          }\r\n\r\n\r\n          actual override fun containsKey(key: K): Boolean =
map.containsKey(key)\r\n\r\n          actual override fun containsValue(value: V): Boolean {\r\n          var node:
ChainEntry<K, V> = head ?: return false\r\n          do {\r\n          if (node.value == value) {\r\n          return
true\r\n          }\r\n          node = node.next!!\r\n          } while (node !== head)\r\n          return false\r\n          }\r\n\r\n\r\n          actual
override fun createEntrySet(): MutableSet<MutableMap.MutableEntry<K, V>> = EntrySet()\r\n\r\n          actual
override operator fun get(key: K): V? = map.get(key)?.value\r\n\r\n          actual override fun put(key: K, value: V): V?
{\r\n          val old = map.get(key)\r\n          if (old == null) {\r\n          val newEntry = ChainEntry(key, value)\r\n
map.put(key, newEntry)\r\n          newEntry.addToEnd()\r\n          return null\r\n          } else {\r\n          return
old.setValue(value)\r\n          }\r\n          }\r\n\r\n\r\n          actual override fun remove(key: K): V? {\r\n          val entry =

```

```

map.remove(key)\r\n    if (entry != null) {\r\n        entry.remove()\r\n        return entry.value\r\n    }\r\n
return null\r\n }\r\n\r\n actual override val size: Int get() = map.size\r\n\r\n}\r\n\r\n/*\r\n * Constructs the
specialized implementation of [LinkedHashMap] with [String] keys, which stores the keys as properties of\r\n * JS
object without hashing them.\r\n */\r\npublic fun <V> linkedStringMapOf(vararg pairs: Pair<String, V>):
LinkedHashMap<String, V> {\r\n    return LinkedHashMap<String, V>(stringMapOf<Any>()).apply { putAll(pairs)
}\r\n}
", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of
this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*/\r\n\r\n * Based on GWT LinkedHashMap\r\n * Copyright 2008 Google Inc.\r\n */\r\n\r\n\r\npackage
kotlin.collections\r\n\r\n/*\r\n * The implementation of the [MutableSet] interface, backed by a [LinkedHashMap]
instance.\r\n */\r\n * This implementation preserves the insertion order of elements during the iteration.\r\n
*/\r\n\r\npublic actual open class LinkedHashMap<E> : HashSet<E>, MutableSet<E> {\r\n\r\n    internal
constructor(map: LinkedHashMap<E, Any>) : super(map)\r\n\r\n    /*\r\n     * Constructs a new empty
[LinkedHashSet].\r\n     */\r\n    actual constructor() : super(LinkedHashMap<E, Any>())\r\n\r\n    /*\r\n     *
Constructs a new [LinkedHashSet] filled with the elements of the specified collection.\r\n     */\r\n    actual
constructor(elements: Collection<E>) : super(LinkedHashMap<E, Any>()) {\r\n        addAll(elements)\r\n
}\r\n\r\n    /*\r\n     * Constructs a new empty [LinkedHashSet].\r\n     */\r\n     * @param initialCapacity the initial
capacity (ignored)\r\n     * @param loadFactor the load factor (ignored)\r\n     */\r\n     * @throws
IllegalArgumentExcepion if the initial capacity or load factor are negative\r\n     */\r\n
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\n    actual
constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : super(LinkedHashMap<E, Any>(initialCapacity,
loadFactor))\r\n\r\n    actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)\r\n\r\n\r\n    public override
fun clone(): Any {\r\n\r\n        return LinkedHashMap(this)\r\n\r\n    }\r\n\r\n\r\n    /*\r\n     * Creates a new instance of
the specialized implementation of [LinkedHashSet] with the specified [String] elements,\r\n     * which elements the
keys as properties of JS object without hashing them.\r\n     */\r\n\r\n    public fun linkedStringSetOf(vararg elements:
String): LinkedHashMap<String> {\r\n        return LinkedHashMap(linkedStringMapOf<Any>()).apply {
addAll(elements) }\r\n    }\r\n}
", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\n\r\npackage kotlin\r\n\r\n\r\nimport kotlin.contracts.*\r\n\r\n\r\n@Deprecated("Do
not use Synchronized annotation in pure Kotlin/JS code", level = DeprecationLevel.ERROR)\r\n\r\npublic typealias
Synchronized = kotlin.jvm.Synchronized\r\n\r\n\r\n@Deprecated("Do not use Volatile annotation in pure Kotlin/JS
code", level = DeprecationLevel.ERROR)\r\n\r\npublic typealias Volatile =
kotlin.jvm.Volatile\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic actual inline fun <R> synchronized(lock: Any,
block: () -> R): R {\r\n    contract {\r\n        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\r\n    }\r\n
return block()\r\n}\r\n}
", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\n\r\npackage kotlin.io\r\n\r\n\r\ninternal abstract class BaseOutput {\r\n    open fun
println() {\r\n        print("\n")\r\n    }\r\n\r\n    open fun println(message: Any?) {\r\n        print(message)\r\n
println()\r\n    }\r\n\r\n    abstract fun print(message: Any?)\r\n\r\n    open fun flush() {} }\r\n\r\n\r\n\r\n    /* JsName used
to make the declaration available outside of module to test it */\r\n    @JsName("NodeJsOutput")\r\n\r\n    internal class
NodeJsOutput(val outputStream: dynamic) : BaseOutput() {\r\n        override fun print(message: Any?) {\r\n            //
TODO: Using local variable because of bug in block decomposition lowering in IR backend\r\n            val
messageString = String(message)\r\n            outputStream.write(messageString)\r\n        }\r\n\r\n\r\n\r\n    /* JsName used to
make the declaration available outside of module to test it */\r\n    @JsName("OutputToConsoleLog")\r\n\r\n    internal class
OutputToConsoleLog : BaseOutput() {\r\n        override fun print(message: Any?) {\r\n
console.log(message)\r\n        }\r\n\r\n        override fun println(message: Any?) {\r\n            console.log(message)\r\n
}\r\n\r\n        override fun println() {\r\n            console.log("")\r\n        }\r\n\r\n\r\n\r\n    /* JsName used to make the
declaration available outside of module to test it and use at try.kotl.in
*/\r\n    @JsName("BufferedOutput")\r\n\r\n    internal open class BufferedOutput : BaseOutput() {\r\n        var buffer =

```

```

override fun print(message: Any?) {
    buffer += String(message)
}

override fun flush() {
    buffer = ""
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("BufferedOutputToConsoleLog")
internal class BufferedOutputToConsoleLog : BufferedOutput() {
    override fun print(message: Any?) {
        var s = String(message)
        val i = s.lastIndexOf('\n')
        if (i >= 0) {
            buffer += s.substring(0, i)
            flush()
            s = s.substring(i + 1)
        }
        buffer += s
    }

    override fun flush() {
        console.log(buffer)
        buffer = ""
    }
}

/** JsName used to make the declaration available outside of module to test it and use at try.kotl.in */
@JsName("output")
internal var output = run {
    val isNode: Boolean = js("typeof process !== 'undefined' && process.versions && !process.versions.node")
    if (isNode) NodeJsOutput(js("process.stdout")) else
    BufferedOutputToConsoleLog()
}

@kotlin.internal.InlineOnly
private inline fun String(value: Any?): String = js("String")(value)

/** Prints the line separator to the standard output stream. */
public actual fun println() {
    output.println()
}

/** Prints the given [message] and the line separator to the standard output stream. */
public actual fun println(message: Any?) {
    output.println(message)
}

/** Prints the given [message] to the standard output stream. */
public actual fun print(message: Any?) {
    output.print(message)
}

"/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package kotlin.coroutines.js.internal

import kotlin.coroutines.Continuation
import kotlin.coroutines.EmptyCoroutineContext
import kotlin.coroutines.PublishedApi
import kotlin.coroutines.SinceKotlin("1.3")
internal val EmptyContinuation = Continuation<Any?>(EmptyCoroutineContext) { result ->
    result.getOrThrow()
}

"/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package kotlin.coroutines

import kotlin.coroutines.intrinsics.CoroutineSingletons

import kotlin.coroutines.intrinsics.COROUTINE_SUSPENDED
import kotlin.coroutines.PublishedApi
import kotlin.coroutines.SinceKotlin("1.3")
internal actual class SafeContinuation<in T>
internal actual constructor(
    private val delegate: Continuation<T>,
    initialResult: Any?
) : Continuation<T> {
    @PublishedApi
    internal actual constructor(delegate: Continuation<T>) : this(delegate, UNDECIDED)

    public actual override val context: CoroutineContext
    get() = delegate.context

    private var result: Any? = initialResult

    public actual override fun resumeWith(result: Result<T>) {
        val cur = this.result
        when {
            cur === UNDECIDED -> {
                this.result = result.value
            }
            cur === COROUTINE_SUSPENDED -> {
                this.result = RESUMED
                delegate.resumeWith(result)
            }
            else -> throw
            IllegalStateException("Already resumed")
        }
    }

    @PublishedApi
    internal actual fun getOrThrow(): Any? {
        if (result === UNDECIDED) {
            result = COROUTINE_SUSPENDED
            return COROUTINE_SUSPENDED
        }
        val result = this.result
        return when {
            result === RESUMED -> COROUTINE_SUSPENDED // already called continuation, indicate
            COROUTINE_SUSPENDED upstream
            result is Result.Failure -> throw result.exception
            else -> result // either COROUTINE_SUSPENDED or data
        }
    }

}

"/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package kotlin.js

/**
 * Exposes the [Date API](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date) to Kotlin.
 */
@Suppress("NOT_DOCUMENTED")
public external class Date() {
    public constructor(milliseconds: Number)
    public constructor(dateString: String)
    public constructor(year: Int, month: Int)
    public constructor(year: Int, month: Int, day: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int, millisecond: Number)
}

```

```

fun getDate(): Int public fun getDay(): Int public fun getFullYear(): Int public fun
getHours(): Int public fun getMilliseconds(): Int public fun getMinutes(): Int public fun
getMonth(): Int public fun getSeconds(): Int public fun getTime(): Double public fun
getTimezoneOffset(): Int public fun getUTCDate(): Int public fun getUTCDay(): Int public fun
public fun getUTCFullYear(): Int public fun getUTCHours(): Int public fun
getUTCMilliseconds(): Int public fun getUTCMinutes(): Int public fun getUTCMonth(): Int public fun
public fun getUTCSeconds(): Int public fun toDate(): String public fun toISOString():
String public fun toJSON(): Json public fun toLocaleDateString(locales: Array<String> =
definedExternally, options: LocaleOptions = definedExternally): String public fun
toLocaleDateString(locales: String, options: LocaleOptions = definedExternally): String public fun
toLocaleString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally):
String public fun toLocaleString(locales: String, options: LocaleOptions = definedExternally): String public fun
public fun toLocaleTimeString(locales: Array<String> = definedExternally, options: LocaleOptions =
definedExternally): String public fun toLocaleTimeString(locales: String, options: LocaleOptions =
definedExternally): String public fun toTimeString(): String public fun toUTCString():
String public companion object { public fun now(): Double public fun parse(dateString:
String): Double public fun UTC(year: Int, month: Int): Double public fun UTC(year: Int,
month: Int, day: Int): Double public fun UTC(year: Int, month: Int, day: Int, hour: Int): Double public fun
public fun UTC(year: Int, month: Int, day: Int, hour: Int, minute: Int): Double public fun UTC(year: Int,
month: Int, day: Int, hour: Int, minute: Int, second: Int): Double public fun UTC(year: Int, month: Int,
day: Int, hour: Int, minute: Int, second: Int, millisecond: Number): Double } public interface
LocaleOptions { public var localeMatcher: String? public var timeZone: String? public var
public var hour12: Boolean public var formatMatcher: String? public var weekday:
String? public var era: String? public var year: String? public var month:
String? public var day: String? public var hour: String? public var minute:
String? public var second: String? public var timeZoneName: String?
} public inline fun dateLocaleOptions(init: Date.LocaleOptions.() -> Unit): Date.LocaleOptions {
val result = js("new Object()").unsafeCast<Date.LocaleOptions>() init(result) return result
}
* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source
code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
*/
package
kotlin.dom
import org.w3c.dom.*
* Creates a new element with the specified [name].
* The element is initialized with the specified [init] function.
public fun Document.createElement(name:
String, init: Element.() -> Unit): Element = createElement(name).apply(init)
* Appends a newly
created element with the specified [name] to this element.
* The element is initialized with the specified
[init] function.
public fun Element.appendChild(name: String, init: Element.() -> Unit): Element =
ownerDocument!!.createElement(name, init).also { appendChild(it) }
* Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.
*/
package kotlin.dom
import org.w3c.dom.*
* Returns true if the element has the given CSS class style in its 'class' attribute
public fun
Element.hasClass(cssClass: String): Boolean =
className.matches("\\s*(^|\\s+)$cssClass(\\s+|$)".toRegex())
* Adds CSS class to element.
Has no effect if all specified classes are already in class attribute of the element
* @return true if at least
one class has been added
public fun Element.addClass(vararg cssClasses: String): Boolean {
val
missingClasses = cssClasses.filterNot { hasClass(it) }
if (missingClasses.isNotEmpty()) {
val
presentClasses = className.trim()
className = buildString {
append(presentClasses)
if
(!presentClasses.isEmpty())
append(" ")
missingClasses.forEach {
append(it)
}
}
return true
}
return false
}
* Removes all [cssClasses] from element. Has
no effect if all specified classes are missing in class attribute of the element
* @return true if at least one

```



```

org.w3c.dom\n\npublic external interface ItemArrayLike<out T> {\n    val length: Int\n    fun item(index: Int): T?\n}\n\n/**\n * Returns the view of this `ItemArrayLike<T>` collection as `List<T>`\n */\n\npublic fun <T> ItemArrayLike<T>.asList(): List<T> = object : AbstractList<T>() {\n    override val size: Int get() = this@asList.length\n\n    override fun get(index: Int): T = when (index) {\n        in 0..lastIndex -> this@asList.item(index).unsafeCast<T>()\n        else -> throw IndexOutOfBoundsException("index $index is not in range [0..$lastIndex]")\n    }\n}\n\n"/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.dom\n\nimport org.w3c.dom.*\n\n/** Removes all the children from this node. */\n\npublic fun Node.clear() {\n    while (hasChildNodes()) {\n        removeChild(firstChild!)\n    }\n}\n\n/**\n * Creates text node and append it to the element.\n */\n\n@return this element\n */\n\nfun Element.appendText(text: String): Element {\n    appendChild(ownerDocument!!.createTextNode(text))\n    return this\n}\n\n"/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin.js\n\n/**\n * Reinterprets this value as a value of the [dynamic type](/docs/reference/dynamic-type.html).\n */\n\n@kotlin.internal.InlineOnly\npublic inline fun Any?.asDynamic(): dynamic = this\n\n/**\n * Reinterprets this value as a value of the specified type [T] without any actual type checking.\n */\n\n@kotlin.internal.InlineOnly\npublic inline fun <T> Any?.unsafeCast(): @kotlin.internal.NoInfer T = this.asDynamic()\n\n/**\n * Reinterprets this `dynamic` value as a value of the specified type [T] without any actual type checking.\n */\n\n@kotlin.internal.DynamicExtension\n@JsName("unsafeCastDynamic")\n@kotlin.internal.InlineOnly\npublic inline fun <T> dynamic.unsafeCast(): @kotlin.internal.NoInfer T = this\n\n/**\n * Allows to iterate this `dynamic` object in the following cases:\n * - when it has an `iterator` function,\n * - when it is an array\n * - when it is an instance of [kotlin.collections.Iterable]\n */\n\n@kotlin.internal.DynamicExtension\npublic operator fun dynamic.iterator(): Iterator<dynamic> {\n    val r: Any? = this\n    return when {\n        this["iterator"] != null -> this["iterator"]()\n        isArrayish(r) -> r.unsafeCast<Array<*>>().iterator()\n        else -> (r as Iterable<*>).iterator()\n    }\n}\n\n"/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n\npackage kotlin\n\n// NOTE: Do not author your exceptions as they are written in this file, instead use this template:\n\npublic open class MyException : Exception {\n    constructor() : super()\n    constructor(message: String?) : super(message)\n    constructor(message: String?, cause: Throwable?) : super(message, cause)\n    constructor(cause: Throwable?) : super(cause)\n}\n\n// TODO: remove workarounds for KT-22053 from direct Throwable inheritors\n// TODO: remove primary constructors, make all secondary KT-22055\n\n@Suppress("USELESS_ELVIS_RIGHT_IS_NULL")\npublic actual open class Error actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null) {\n    actual constructor() : this(null, null)\n    Error::class.js.asDynamic().call(this, null, null)\n    actual constructor(message: String?) : this(message, null)\n    Error::class.js.asDynamic().call(this, message, null)\n    actual constructor(cause: Throwable?) : this(undefiend, cause)\n    Error::class.js.asDynamic().call(this, undefiend, cause)\n}\n\n@Suppress("USELESS_ELVIS_RIGHT_IS_NULL")\npublic actual open class Exception actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null) {\n    actual constructor() : this(null, null)\n    Exception::class.js.asDynamic().call(this, null, null)\n    actual constructor(message: String?) : this(message, null)\n    Exception::class.js.asDynamic().call(this, message, null)\n    actual constructor(cause: Throwable?) : this(undefiend, cause)\n    Exception::class.js.asDynamic().call(this, undefiend, cause)\n}\n\npublic actual open class RuntimeException actual constructor(message: String?, cause: Throwable?) : Exception(message, cause) {\n    actual constructor() : this(null, null)\n    actual constructor(message: String?) : this(message, null)\n    actual constructor(cause: Throwable?) : this(undefiend, cause)\n}\n\npublic actual open class

```

```

IllegalArgumentException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message,
cause) {\r\n  actual constructor() : this(null, null)\r\n  actual constructor(message: String?) : this(message, null)\r\n
  actual constructor(cause: Throwable?) : this(undefiend, cause)\r\n}\r\n\r\npublic actual open class
IllegalStateException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
{\r\n  actual constructor() : this(null, null)\r\n  actual constructor(message: String?) : this(message, null)\r\n
  actual constructor(cause: Throwable?) : this(undefiend, cause)\r\n}\r\n\r\npublic actual open class
IndexOutOfBoundsException actual constructor(message: String?) : RuntimeException(message) {\r\n  actual
constructor() : this(null)\r\n}\r\n\r\npublic actual open class ConcurrentModificationException actual
constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {\r\n  actual constructor() :
this(null, null)\r\n  actual constructor(message: String?) : this(message, null)\r\n  actual constructor(cause:
Throwable?) : this(undefiend, cause)\r\n}\r\n\r\npublic actual open class UnsupportedOperationException actual
constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {\r\n  actual constructor() :
this(null, null)\r\n  actual constructor(message: String?) : this(message, null)\r\n  actual constructor(cause:
Throwable?) : this(undefiend, cause)\r\n}\r\n\r\n\r\npublic actual open class NumberFormatException actual
constructor(message: String?) : IllegalArgumentException(message) {\r\n  actual constructor() :
this(null)\r\n}\r\n\r\n\r\npublic actual open class NullPointerException actual constructor(message: String?) :
RuntimeException(message) {\r\n  actual constructor() : this(null)\r\n}\r\n\r\n\r\npublic actual open class
ClassCastException actual constructor(message: String?) : RuntimeException(message) {\r\n  actual constructor() :
this(null)\r\n}\r\n\r\n\r\npublic actual open class AssertionError private constructor(message: String?, cause:
Throwable?) : Error(message, cause) {\r\n  actual constructor() : this(null)\r\n  constructor(message: String?) :
this(message, null)\r\n  actual constructor(message: Any?) : this(message.toString(), message as?
Throwable)\r\n}\r\n\r\n\r\npublic actual open class NoSuchElementException actual constructor(message: String?) :
RuntimeException(message) {\r\n  actual constructor() : this(null)\r\n}\r\n\r\n\r\n@SinceKotlin("1.3")\r\n\r\npublic
actual open class ArithmeticException actual constructor(message: String?) : RuntimeException(message) {\r\n
  actual constructor() : this(null)\r\n}\r\n\r\n\r\npublic actual open class NoWhenBranchMatchedException actual
constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {\r\n  actual constructor() :
this(null, null)\r\n  actual constructor(message: String?) : this(message, null)\r\n  actual constructor(cause:
Throwable?) : this(undefiend, cause)\r\n}\r\n\r\n\r\npublic actual open class UninitializedPropertyAccessException
actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {\r\n  actual
constructor() : this(null, null)\r\n  actual constructor(message: String?) : this(message, null)\r\n  actual
constructor(cause: Throwable?) : this(undefiend, cause)\r\n}\r\n\r\n", /*\r\n * Copyright 2010-2018 JetBrains s.r.o. and
Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that
can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n\r\n// a package is omitted to get declarations directly under
the module\r\n\r\n\r\n@JsName("throwNPE")\r\n\r\ninternal fun throwNPE(message: String) {\r\n  throw
NullPointerException(message)\r\n}\r\n\r\n\r\n\r\n@JsName("throwCCE")\r\n\r\ninternal fun throwCCE() {\r\n  throw
ClassCastException("Illegal cast")\r\n}\r\n\r\n\r\n\r\n@JsName("throwISE")\r\n\r\ninternal fun throwISE(message: String)
{\r\n  throw IllegalStateException(message)\r\n}\r\n\r\n\r\n\r\n@JsName("throwUPAE")\r\n\r\ninternal fun
throwUPAE(propertyName: String) {\r\n  throw UninitializedPropertyAccessException("lateinit property
${propertyName} has not been initialized")\r\n}\r\n\r\n\r\n", /*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n
\r\n\r\n\r\n\r\n@file:kotlin.jvm.JvmName("GroupingKt")\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n\r\n\r\npackage
kotlin.collections\r\n\r\n\r\n**\r\n * Represents a source of elements with a [keyOf] function, which can be applied to
each element to get its key.\r\n\r\n * A [Grouping] structure serves as an intermediate step in group-and-fold
operations:\r\n\r\n * they group elements by their keys and then fold each group with some aggregating operation.\r\n\r\n
*\r\n * It is created by attaching `keySelector: (T) -> K` function to a source of elements.\r\n\r\n * To get an instance of
[Grouping] use one of `groupingBy` extension functions:\r\n\r\n * - [Iterable.groupingBy]\r\n\r\n * -
[Sequence.groupingBy]\r\n\r\n * - [Array.groupingBy]\r\n\r\n * - [CharSequence.groupingBy]\r\n\r\n * For the list of

```

group-and-fold operations available, see the [extension functions](#extension-functions) for `Grouping`.

```

*@\r\n@SinceKotlin("1.1")\r\npublic interface Grouping<T, out K> {\r\n    /** Returns an [Iterator] over the
elements of the source of this grouping. */\r\n    fun sourceIterator(): Iterator<T>\r\n    /** Extracts the key of an
[element]. */\r\n    fun keyOf(element: T): K\r\n}\r\n\r\n/**\r\n * Groups elements from the [Grouping] source by
key and applies [operation] to the elements of each group sequentially,\r\n * passing the previously accumulated
value and the current element as arguments, and stores the results in a new map.\r\n * *\r\n * The key for each element
is provided by the [Grouping.keyOf] function.\r\n * *\r\n * @param operation function is invoked on each element
with the following parameters:\r\n * - `key`: the key of the group this element belongs to;\r\n * - `accumulator`: the
current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;\r\n *
- `element`: the element from the source being aggregated;\r\n * - `first`: indicates whether it's the first `element`
encountered in the group.\r\n * *\r\n * @return a [Map] associating the key of each group with the result of
aggregation of the group elements.\r\n * @sample samples.collections.Grouping.aggregateByRadix\r\n
*\r\n@SinceKotlin("1.1")\r\npublic inline fun <T, K, R> Grouping<T, K>.aggregate(\r\n    operation: (key: K,
accumulator: R?, element: T, first: Boolean) -> R\r\n): Map<K, R> {\r\n    return aggregateTo(mutableMapOf<K,
R>(), operation)\r\n}\r\n\r\n/**\r\n * Groups elements from the [Grouping] source by key and applies [operation] to
the elements of each group sequentially,\r\n * passing the previously accumulated value and the current element as
arguments,\r\n * and stores the results in the given [destination] map.\r\n * *\r\n * The key for each element is
provided by the [Grouping.keyOf] function.\r\n * *\r\n * @param operation a function that is invoked on each element
with the following parameters:\r\n * - `key`: the key of the group this element belongs to;\r\n * - `accumulator`: the
current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;\r\n *
- `element`: the element from the source being aggregated;\r\n * - `first`: indicates whether it's the first `element`
encountered in the group.\r\n * *\r\n * If the [destination] map already has a value corresponding to some key,\r\n *
then the elements being aggregated for that key are never considered as `first`.\r\n * *\r\n * @return the [destination]
map associating the key of each group with the result of aggregation of the group elements.\r\n * @sample
samples.collections.Grouping.aggregateByRadixTo\r\n
*\r\n@SinceKotlin("1.1")\r\npublic inline fun <T, K, R, M
: MutableMap<in K, R>> Grouping<T, K>.aggregateTo(\r\n    destination: M,\r\n    operation: (key: K,
accumulator: R?, element: T, first: Boolean) -> R\r\n): M {\r\n    for (e in this.sourceIterator()) {\r\n        val key =
keyOf(e)\r\n        val accumulator = destination[key]\r\n        destination[key] = operation(key, accumulator, e,
accumulator == null && !destination.containsKey(key))\r\n    }\r\n    return destination\r\n}\r\n\r\n/**\r\n * Groups
elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially,\r\n *
passing the previously accumulated value and the current element as arguments, and stores the results in a new
map.\r\n * *\r\n * An initial value of accumulator is provided by [initialValueSelector] function.\r\n * *\r\n * @param
initialValueSelector a function that provides an initial value of accumulator for each group.\r\n * *\r\n * It's invoked
with parameters:\r\n * - `key`: the key of the group;\r\n * - `element`: the first element being encountered in that
group.\r\n * *\r\n * @param operation a function that is invoked on each element with the following parameters:\r\n *
- `key`: the key of the group this element belongs to;\r\n * - `accumulator`: the current value of the accumulator of
the group;\r\n * - `element`: the element from the source being accumulated.\r\n * *\r\n * @return a [Map]
associating the key of each group with the result of accumulating the group elements.\r\n * @sample
samples.collections.Grouping.foldByEvenLengthWithComputedInitialValue\r\n
*\r\n@SinceKotlin("1.1")\r\npublic inline fun <T, K, R> Grouping<T, K>.fold(\r\n    initialValueSelector: (key:
K, element: T) -> R,\r\n    operation: (key: K, accumulator: R, element: T) -> R\r\n): Map<K, R> =\r\n    @Suppress("UNCHECKED_CAST")\r\n    aggregate { key, acc, e, first -> operation(key, if (first)
initialValueSelector(key, e) else acc as R, e) }\r\n}\r\n\r\n/**\r\n * Groups elements from the [Grouping] source by key
and applies [operation] to the elements of each group sequentially,\r\n * passing the previously accumulated value
and the current element as arguments,\r\n * and stores the results in the given [destination] map.\r\n * *\r\n * An initial
value of accumulator is provided by [initialValueSelector] function.\r\n * *\r\n * @param initialValueSelector a
function that provides an initial value of accumulator for each group.\r\n * *\r\n * It's invoked with parameters:\r\n *
- `key`: the key of the group;\r\n * - `element`: the first element being encountered in that group.\r\n * *\r\n * If the

```

[destination] map already has a value corresponding to some key, that value is used as an initial value of the accumulator for that group and the [initialValueSelector] function is not called for that group.

`@param operation` a function that is invoked on each element with the following parameters:

- `key`: the key of the group this element belongs to;
- `accumulator`: the current value of the accumulator of the group;
- `element`: the element from the source being accumulated.

`@return` the [destination] map associating the key of each group with the result of accumulating the group elements.

`@sample`

```
samples.collections.Grouping.foldByEvenLengthWithComputedInitialValueTo
```

```
*\r\n@SinceKotlin("1.1")\r\npublic inline fun <T, K, R, M : MutableMap<in K, R>> Grouping<T, K>.foldTo(\r\n
    destination: M,\r\n    initialValueSelector: (key: K, element: T) -> R,\r\n    operation: (key: K, accumulator: R,
    element: T) -> R\r\n): M =\r\n    @Suppress("UNCHECKED_CAST")\r\n    aggregateTo(destination) { key, acc, e,
    first -> operation(key, if (first) initialValueSelector(key, e) else acc as R, e) }\r\n\r\n/**\r\n * Groups elements
    from the [Grouping] source by key and applies [operation] to the elements of each group sequentially,\r\n * passing
    the previously accumulated value and the current element as arguments, and stores the results in a new map.\r\n *
    An initial value of accumulator is the same [initialValue] for each group.\r\n * @param operation a function
    that is invoked on each element with the following parameters:\r\n * - `accumulator`: the current value of the
    accumulator of the group;\r\n * - `element`: the element from the source being accumulated.\r\n * @return a
    [Map] associating the key of each group with the result of accumulating the group elements.\r\n * @sample
    samples.collections.Grouping.foldByEvenLengthWithConstantInitialValue\r\n
    *\r\n@SinceKotlin("1.1")\r\npublic inline fun <T, K, R> Grouping<T, K>.fold(\r\n    initialValue: R,\r\n
    operation: (accumulator: R, element: T) -> R\r\n): Map<K, R> =\r\n    @Suppress("UNCHECKED_CAST")\r\n
    aggregate { _, acc, e, first -> operation(if (first) initialValue else acc as R, e) }\r\n\r\n/**\r\n * Groups elements from
    the [Grouping] source by key and applies [operation] to the elements of each group sequentially,\r\n * passing the
    previously accumulated value and the current element as arguments,\r\n * and stores the results in the given
    [destination] map.\r\n * An initial value of accumulator is the same [initialValue] for each group.\r\n * If the
    [destination] map already has a value corresponding to the key of some group,\r\n * that value is used as an initial
    value of the accumulator for that group.\r\n * @param operation a function that is invoked on each element
    with the following parameters:\r\n * - `accumulator`: the current value of the accumulator of the group;\r\n * -
    `element`: the element from the source being accumulated.\r\n * @return the [destination] map associating the
    key of each group with the result of accumulating the group elements.\r\n * @sample
    samples.collections.Grouping.foldByEvenLengthWithConstantInitialValueTo\r\n
    *\r\n@SinceKotlin("1.1")\r\npublic inline fun <T, K, R, M : MutableMap<in K, R>> Grouping<T, K>.foldTo(\r\n
    destination: M,\r\n    initialValue: R,\r\n    operation: (accumulator: R, element: T) -> R\r\n): M =\r\n
    @Suppress("UNCHECKED_CAST")\r\n    aggregateTo(destination) { _, acc, e, first -> operation(if (first)
    initialValue else acc as R, e) }\r\n\r\n/**\r\n * Groups elements from the [Grouping] source by key and applies
    the reducing [operation] to the elements of each group\r\n * sequentially starting from the second element of the
    group,\r\n * passing the previously accumulated value and the current element as arguments,\r\n * and stores the
    results in a new map.\r\n * An initial value of accumulator is the first element of the group.\r\n * @param
    operation a function that is invoked on each subsequent element of the group with the following parameters:\r\n *
    - `key`: the key of the group this element belongs to;\r\n * - `accumulator`: the current value of the accumulator of
    the group;\r\n * - `element`: the element from the source being accumulated.\r\n * @return a [Map]
    associating the key of each group with the result of accumulating the group elements.\r\n * @sample
    samples.collections.Grouping.reduceByMaxVowels\r\n *\r\n@SinceKotlin("1.1")\r\npublic inline fun <S, T : S,
    K> Grouping<T, K>.reduce(\r\n    operation: (key: K, accumulator: S, element: T) -> S\r\n): Map<K, S> =\r\n
    @Suppress("UNCHECKED_CAST")\r\n    aggregate { key, acc, e, first ->\r\n        @Suppress("UNCHECKED_CAST")\r\n
        if (first) e else operation(key, acc as S, e)\r\n    }\r\n\r\n/**\r\n * Groups elements from the [Grouping] source by key and applies the reducing
    [operation] to the elements of each group\r\n * sequentially starting from the second element of the group,\r\n *
    passing the previously accumulated value and the current element as arguments,\r\n * and stores the results in the
    given [destination] map.\r\n * An initial value of accumulator is the first element of the group.\r\n * If the
```

[destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the accumulator for that group and the first element of that group is also subjected to the [operation].

@param operation a function that is invoked on each subsequent element of the group with the following parameters:

- accumulator: the current value of the accumulator of the group;
- element: the element from the source being folded;

@return the [destination] map associating the key of each group with the result of accumulating the group elements.

@sample

```

samples.collections.Grouping.reduceByMaxVowelsTo
@SinceKotlin("1.1")
public inline fun <S, T : S, K, M : MutableMap<in K, S>> Grouping<T, K>.reduceTo(
    destination: M,
    operation: (key: K, accumulator: S, element: T) -> S): M =
    aggregateTo(destination) { key, acc, e, first ->
        @Suppress("UNCHECKED_CAST")
        if (first) e else operation(key, acc as S, e)
    }

```

Groups elements from the [Grouping] source by key and counts elements in each group to the given [destination] map.

If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the counter for that group.

@return the [destination] map associating the key of each group with the count of elements in the group.

@sample

```

samples.collections.Grouping.groupingByEachCount
@SinceKotlin("1.1")
public fun <T, K, M : MutableMap<in K, Int>> Grouping<T, K>.eachCountTo(
    destination: M): M =
    foldTo(destination, 0) { acc, _ -> acc + 1 }

```

Groups elements from the [Grouping] source by key and sums values provided by the [valueSelector] function for elements in each group to the given [destination] map.

If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the sum for that group.

@return the [destination] map associating the key of each group with the sum of elements in the group.

```

@SinceKotlin("1.1")
public inline fun <T, K, M : MutableMap<in K, Int>> Grouping<T, K>.eachSumOf(
    destination: M, valueSelector: (T) -> Int): M =
    foldTo(destination, 0) { acc, e -> acc + valueSelector(e) }
// TODO: sum by long and by double overloads
public inline fun <T, K, M : MutableMap<in K, Long>> Grouping<T, K>.sumEachByLongTo(
    destination: M, valueSelector: (T) -> Long): M =
    foldTo(destination, 0L) { acc, e -> acc + valueSelector(e) }
public inline fun <T, K> Grouping<T, K>.sumEachByLong(
    valueSelector: (T) -> Long): Map<K, Long> =
    fold(0L) { acc, e -> acc + valueSelector(e) }
public inline fun <T, K, M : MutableMap<in K, Double>> Grouping<T, K>.sumEachByDoubleTo(
    destination: M, valueSelector: (T) -> Double): M =
    foldTo(destination, 0.0) { acc, e -> acc + valueSelector(e) }
public inline fun <T, K> Grouping<T, K>.sumEachByDouble(
    valueSelector: (T) -> Double): Map<K, Double> =
    fold(0.0) { acc, e -> acc + valueSelector(e) }

```

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

package kotlin.collections

Groups elements from the [Grouping] source by key and counts elements in each group.

@return a [Map] associating the key of each group with the count of element in the group.

```

@SinceKotlin("1.1")
public actual fun <T, K> Grouping<T, K>.eachCount(): Map<K, Int> =
    fold(0) { acc, _ -> acc + 1 }

```

Groups elements from the [Grouping] source by key and sums values provided by the [valueSelector] function for elements in each group.

@return a [Map] associating the key of each group with the count of element in the group.

```

@SinceKotlin("1.1")
public inline fun <T, K> Grouping<T, K>.eachSumOf(
    valueSelector: (T) -> Int): Map<K, Int> =
    fold(0) { acc, e -> acc + valueSelector(e) }

```

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

package kotlin.js

Exposes the JavaScript [Math object](https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Global_Objects/Math) to Kotlin.

@Deprecated("Use top-level functions from kotlin.math package instead.", level = DeprecationLevel.WARNING)

```

public external object Math {
    @Deprecated("Use kotlin.math.PI instead.", ReplaceWith("PI", "kotlin.math.PI"), level = DeprecationLevel.ERROR)
    public val PI: Double
    @Deprecated("Use Random.nextDouble instead", ReplaceWith("kotlin.random.Random.nextDouble()", "kotlin.random.Random"), level = DeprecationLevel.WARNING)
    public fun random(): Double

```

```

@Deprecated("Use kotlin.math.abs instead.", ReplaceWith("abs(value)", "kotlin.math.abs"), level =
DeprecationLevel.ERROR)\r\n public fun abs(value: Double): Double\r\n @Deprecated("Use kotlin.math.acos
instead.", ReplaceWith("acos(value)", "kotlin.math.acos"), level = DeprecationLevel.ERROR)\r\n public fun
acos(value: Double): Double\r\n @Deprecated("Use kotlin.math.asin instead.", ReplaceWith("asin(value)",
"kotlin.math.asin"), level = DeprecationLevel.ERROR)\r\n public fun asin(value: Double): Double\r\n
@Deprecated("Use kotlin.math.atan instead.", ReplaceWith("atan(value)", "kotlin.math.atan"), level =
DeprecationLevel.ERROR)\r\n public fun atan(value: Double): Double\r\n @Deprecated("Use
kotlin.math.atan2 instead.", ReplaceWith("atan2(y, x)", "kotlin.math.atan2"), level =
DeprecationLevel.ERROR)\r\n public fun atan2(y: Double, x: Double): Double\r\n @Deprecated("Use
kotlin.math.cos instead.", ReplaceWith("cos(value)", "kotlin.math.cos"), level = DeprecationLevel.ERROR)\r\n
public fun cos(value: Double): Double\r\n @Deprecated("Use kotlin.math.sin instead.",
ReplaceWith("sin(value)", "kotlin.math.sin"), level = DeprecationLevel.ERROR)\r\n public fun sin(value:
Double): Double\r\n @Deprecated("Use kotlin.math.exp instead.", ReplaceWith("exp(value)",
"kotlin.math.exp"), level = DeprecationLevel.ERROR)\r\n public fun exp(value: Double): Double\r\n
@Deprecated("Use maxOf or kotlin.math.max instead", level = DeprecationLevel.ERROR)\r\n public fun
max(vararg values: Int): Int\r\n @Deprecated("Use maxOf or kotlin.math.max instead", level =
DeprecationLevel.ERROR)\r\n public fun max(vararg values: Float): Float\r\n @Deprecated("Use maxOf or
kotlin.math.max instead", level = DeprecationLevel.ERROR)\r\n public fun max(vararg values: Double):
Double\r\n @Deprecated("Use minOf or kotlin.math.min instead", level = DeprecationLevel.ERROR)\r\n
public fun min(vararg values: Int): Int\r\n @Deprecated("Use minOf or kotlin.math.min instead", level =
DeprecationLevel.ERROR)\r\n public fun min(vararg values: Float): Float\r\n @Deprecated("Use minOf or
kotlin.math.min instead", level = DeprecationLevel.ERROR)\r\n public fun min(vararg values: Double):
Double\r\n @Deprecated("Use kotlin.math.sqrt instead.", ReplaceWith("sqrt(value)", "kotlin.math.sqrt"), level
= DeprecationLevel.ERROR)\r\n public fun sqrt(value: Double): Double\r\n @Deprecated("Use kotlin.math.tan
instead.", ReplaceWith("tan(value)", "kotlin.math.tan"), level = DeprecationLevel.ERROR)\r\n public fun
tan(value: Double): Double\r\n @Deprecated("Use kotlin.math.ln instead.", ReplaceWith("ln(value)",
"kotlin.math.ln"), level = DeprecationLevel.ERROR)\r\n public fun log(value: Double): Double\r\n
@Deprecated("Use kotlin.math.pow instead.", ReplaceWith("base.pow(exp)", "kotlin.math.pow"), level =
DeprecationLevel.ERROR)\r\n public fun pow(base: Double, exp: Double): Double\r\n @Deprecated("Use
kotlin.math.round instead.", ReplaceWith("round(value)", "kotlin.math.round"), level =
DeprecationLevel.ERROR)\r\n public fun round(value: Number): Int\r\n @Deprecated("Use kotlin.math.floor
instead.", ReplaceWith("floor(value)", "kotlin.math.floor"), level = DeprecationLevel.ERROR)\r\n public fun
floor(value: Number): Int\r\n @Deprecated("Use kotlin.math.ceil instead.", ReplaceWith("ceil(value)",
"kotlin.math.ceil"), level = DeprecationLevel.ERROR)\r\n public fun ceil(value: Number): Int\r\n\r\n
@PublishedApi\r\n internal fun trunc(value: Number): Double\r\n @PublishedApi\r\n internal fun sign(value:
Number): Double\r\n\r\n @PublishedApi\r\n internal fun sinh(value: Double): Double\r\n @PublishedApi\r\n
internal fun cosh(value: Double): Double\r\n @PublishedApi\r\n internal fun tanh(value: Double): Double\r\n
@PublishedApi\r\n internal fun asinh(value: Double): Double\r\n @PublishedApi\r\n internal fun acosh(value:
Double): Double\r\n @PublishedApi\r\n internal fun atanh(value: Double): Double\r\n\r\n @PublishedApi\r\n
internal fun hypot(x: Double, y: Double): Double\r\n\r\n @PublishedApi\r\n internal fun expm1(value: Double):
Double\r\n\r\n @PublishedApi\r\n internal fun log10(value: Double): Double\r\n @PublishedApi\r\n internal
fun log2(value: Double): Double\r\n @PublishedApi\r\n internal fun log1p(value: Double): Double\r\n\r\n
@PublishedApi\r\n internal fun clz32(value: Int): Int\r\n\r\n\r\n * Returns the smaller of two values.\r\n
*\r\n @Suppress("DEPRECATION")\r\n @Deprecated("Use minOf or kotlin.math.min instead",
ReplaceWith("minOf(a, b)"), level = DeprecationLevel.ERROR)\r\n public fun Math.min(a: Long, b: Long): Long
= if (a <= b) a else b\r\n\r\n * Returns the greater of two values.\r\n
*\r\n @Suppress("DEPRECATION")\r\n @Deprecated("Use maxOf or kotlin.math.max instead",
ReplaceWith("maxOf(a, b)"), level = DeprecationLevel.ERROR)\r\n public fun Math.max(a: Long, b: Long): Long

```

```

= if (a >= b) a else b\r\n", /*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.js\r\n\r\n/**\r\n * An interface for indexing access to a
collection of key-value pairs, where type of key is [String] and type of value is [Any?][Any].\r\n */\r\n\r\npublic
external interface Json {\r\n    /**\r\n     * Calls to the function will be translated to indexing operation (square
brackets) on the receiver with [propertyName] as the argument.\r\n     * \r\n     * E.g. for next code:\r\n     *
```\r\n\r\n    * fun test(j: Json, p: String) = j["prop"] + j.get(p)\r\n    * ```\r\n     * \r\n     * will be generated:\r\n\r\n    * ```js\r\n    * function test(j, p) {\r\n    *     return j["prop"] + j[p];\r\n    * }\r\n    * ```\r\n     * \r\n     */\r\n     * An operator fun
get(propertyName: String): Any?\r\n\r\n * \r\n * Calls of the function will be translated to an assignment of
[value] to the receiver indexed (with square brackets/index operation) with [propertyName].\r\n * \r\n * E.g. for
the following code:\r\n * ```\r\n\r\n * fun test(j: Json, p: String, newValue: Any) {\r\n * j["prop"] =
1\r\n * j.set(p, newValue)\r\n * }\r\n * ```\r\n * \r\n * will be generated:\r\n * ```js\r\n * function
test(j, p, newValue) {\r\n * j["prop"] = 1;\r\n * j[p] = newValue;\r\n * }\r\n * }\r\n * ```\r\n
 * \r\n */\r\n * An operator fun set(propertyName: String, value: Any?): Unit\r\n\r\n * \r\n * Returns a simple JavaScript
object (as [Json]) using provided key-value pairs as names and values of its properties.\r\n */\r\n\r\n public fun
json(vararg pairs: Pair<String, Any?>): Json {\r\n val res: dynamic = js("{}")\r\n for ((name, value) in pairs)
{\r\n res[name] = value\r\n }\r\n return res\r\n\r\n * \r\n * Adds key-value pairs from [other] to
[this].\r\n * \r\n * Returns the original receiver.\r\n */\r\n\r\n public fun Json.add(other: Json): Json {\r\n val keys:
Array<String> = js("Object").keys(other)\r\n for (key in keys) {\r\n if
(other.asDynamic().hasOwnProperty(key)) {\r\n this[key] = other[key];\r\n }\r\n }\r\n return
this\r\n\r\n * \r\n * Exposes the JavaScript [JSON object](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/JSON) to Kotlin.\r\n\r\n * \r\n * @Suppress("NOT_DOCUMENTED")\r\n public external object JSON {\r\n public fun stringify(o: Any?):
String\r\n public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?): String)\r\n public fun
stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)? = definedExternally, space: Int): String\r\n public
fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)? = definedExternally, space: String):
String\r\n public fun stringify(o: Any?, replacer: Array<String>): String\r\n public fun stringify(o: Any?,
replacer: Array<String>, space: Int): String\r\n public fun stringify(o: Any?, replacer: Array<String>, space:
String): String\r\n public fun <T> parse(text: String): T\r\n public fun <T> parse(text: String, reviver: ((key:
String, value: Any?) -> Any?): T)\r\n\r\n * \r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n @file:Suppress("DEPRECATION", "DEPRECATION_ERROR")\r\n\r\n package
kotlin.math\r\n\r\n \r\n \r\n import kotlin.internal.InlineOnly\r\n import kotlin.js.Math as nativeMath\r\n\r\n // region
===== Double Math =====\r\n\r\n * \r\n * Computes
the sine of the angle [x] given in radians.\r\n * \r\n * Special cases:\r\n * - `sin(NaN|+Inf|-Inf)` is `NaN`\r\n
 * \r\n * @SinceKotlin("1.2")\r\n @InlineOnly\r\n public actual inline fun sin(x: Double): Double =
nativeMath.sin(x)\r\n\r\n * \r\n * Computes the cosine of the angle [x] given in radians.\r\n * \r\n * Special cases:\r\n * -
`cos(NaN|+Inf|-Inf)` is `NaN`\r\n * \r\n * @SinceKotlin("1.2")\r\n @InlineOnly\r\n public actual inline fun cos(x:
Double): Double = nativeMath.cos(x)\r\n\r\n * \r\n * Computes the tangent of the angle [x] given in radians.\r\n * \r\n *
Special cases:\r\n * - `tan(NaN|+Inf|-Inf)` is `NaN`\r\n * \r\n * @SinceKotlin("1.2")\r\n @InlineOnly\r\n public
actual inline fun tan(x: Double): Double = nativeMath.tan(x)\r\n\r\n * \r\n * Computes the arc sine of the value
[x];\r\n * \r\n * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.\r\n * \r\n * Special cases:\r\n * -
`asin(x)` is `NaN`, when `abs(x) > 1` or x is `NaN`\r\n * \r\n * @SinceKotlin("1.2")\r\n @InlineOnly\r\n public actual
inline fun asin(x: Double): Double = nativeMath.asin(x)\r\n\r\n * \r\n * Computes the arc cosine of the value [x];\r\n *
\r\n * the returned value is an angle in the range from `0.0` to `PI` radians.\r\n * \r\n * Special cases:\r\n * - `acos(x)` is
`NaN`, when `abs(x) > 1` or x is `NaN`\r\n * \r\n * @SinceKotlin("1.2")\r\n @InlineOnly\r\n public actual inline fun
acos(x: Double): Double = nativeMath.acos(x)\r\n\r\n * \r\n * Computes the arc tangent of the value [x];\r\n * \r\n * the
returned value is an angle in the range from `-PI/2` to `PI/2` radians.\r\n * \r\n * Special cases:\r\n * - `atan(NaN)` is

```

`NaN` Since Kotlin("1.2") Inline Only public actual inline fun atan(x: Double): Double = nativeMath.atan(x) Returns the angle  $\theta$  of the polar coordinates  $(r, \theta)$  that correspond to the rectangular coordinates  $(x, y)$  by computing the arc tangent of the value  $y / x$ ; the returned value is an angle in the range from  $-\pi$  to  $\pi$  radians. Special cases:  $\text{atan2}(0.0, 0.0)$  is  $0.0$ ;  $\text{atan2}(0.0, x)$  is  $0.0$  for  $x > 0$  and  $\pi$  for  $x < 0$ ;  $\text{atan2}(-0.0, x)$  is  $-0.0$  for  $x > 0$  and  $-\pi$  for  $x < 0$ ;  $\text{atan2}(y, +\text{Inf})$  is  $0.0$  for  $0 < y < +\text{Inf}$  and  $-0.0$  for  $-\text{Inf} < y < 0$ ;  $\text{atan2}(y, -\text{Inf})$  is  $\pi$  for  $0 < y < +\text{Inf}$  and  $-\pi$  for  $-\text{Inf} < y < 0$ ;  $\text{atan2}(y, 0.0)$  is  $\pi/2$  for  $y > 0$  and  $-\pi/2$  for  $y < 0$ ;  $\text{atan2}(+\text{Inf}, x)$  is  $\pi/2$  for finite  $x > 0$ ;  $\text{atan2}(-\text{Inf}, x)$  is  $-\pi/2$  for finite  $x$ ;  $\text{atan2}(\text{NaN}, x)$  and  $\text{atan2}(y, \text{NaN})$  is  $\text{NaN}$ .

Since Kotlin("1.2") Inline Only public actual inline fun atan2(y: Double, x: Double): Double = nativeMath.atan2(y, x) Computes the hyperbolic sine of the value  $x$ . Special cases:  $\sinh(\text{NaN})$  is  $\text{NaN}$ ;  $\sinh(+\text{Inf})$  is  $+\text{Inf}$ ;  $\sinh(-\text{Inf})$  is  $-\text{Inf}$ .

Since Kotlin("1.2") Inline Only public actual inline fun sinh(x: Double): Double = nativeMath.sinh(x) Computes the hyperbolic cosine of the value  $x$ . Special cases:  $\cosh(\text{NaN})$  is  $\text{NaN}$ ;  $\cosh(+\text{Inf}-\text{Inf})$  is  $+\text{Inf}$ .

Since Kotlin("1.2") Inline Only public actual inline fun cosh(x: Double): Double = nativeMath.cosh(x) Computes the hyperbolic tangent of the value  $x$ . Special cases:  $\tanh(\text{NaN})$  is  $\text{NaN}$ ;  $\tanh(+\text{Inf})$  is  $1.0$ ;  $\tanh(-\text{Inf})$  is  $-1.0$ .

Since Kotlin("1.2") Inline Only public actual inline fun tanh(x: Double): Double = nativeMath.tanh(x) Computes the inverse hyperbolic sine of the value  $x$ . The returned value is  $y$  such that  $\sinh(y) = x$ . Special cases:  $\text{asinh}(\text{NaN})$  is  $\text{NaN}$ ;  $\text{asinh}(+\text{Inf})$  is  $+\text{Inf}$ ;  $\text{asinh}(-\text{Inf})$  is  $-\text{Inf}$ .

Since Kotlin("1.2") Inline Only public actual inline fun asinh(x: Double): Double = nativeMath.asinh(x) Computes the inverse hyperbolic cosine of the value  $x$ . The returned value is positive  $y$  such that  $\cosh(y) = x$ . Special cases:  $\text{acosh}(\text{NaN})$  is  $\text{NaN}$ ;  $\text{acosh}(x)$  is  $\text{NaN}$  when  $x < 1$ ;  $\text{acosh}(+\text{Inf})$  is  $+\text{Inf}$ .

Since Kotlin("1.2") Inline Only public actual inline fun acosh(x: Double): Double = nativeMath.acosh(x) Computes the inverse hyperbolic tangent of the value  $x$ . The returned value is  $y$  such that  $\tanh(y) = x$ . Special cases:  $\text{atanh}(\text{NaN})$  is  $\text{NaN}$ ;  $\text{atanh}(x)$  is  $\text{NaN}$  when  $x > 1$  or  $x < -1$ ;  $\text{atanh}(1.0)$  is  $+\text{Inf}$ ;  $\text{atanh}(-1.0)$  is  $-\text{Inf}$ .

Since Kotlin("1.2") Inline Only public actual inline fun atanh(x: Double): Double = nativeMath.atanh(x) Computes  $\sqrt{x^2 + y^2}$  without intermediate overflow or underflow. Special cases: returns  $+\text{Inf}$  if any of arguments is infinite; returns  $\text{NaN}$  if any of arguments is  $\text{NaN}$  and the other is not infinite.

Since Kotlin("1.2") Inline Only public actual inline fun hypot(x: Double, y: Double): Double = nativeMath.hypot(x, y) Computes the positive square root of the value  $x$ . Special cases:  $\sqrt{x}$  is  $\text{NaN}$  when  $x < 0$  or  $x$  is  $\text{NaN}$ .

Since Kotlin("1.2") Inline Only public actual inline fun sqrt(x: Double): Double = nativeMath.sqrt(x) Computes Euler's number  $e$  raised to the power of the value  $x$ . Special cases:  $\exp(\text{NaN})$  is  $\text{NaN}$ ;  $\exp(+\text{Inf})$  is  $+\text{Inf}$ ;  $\exp(-\text{Inf})$  is  $0.0$ .

Since Kotlin("1.2") Inline Only public actual inline fun exp(x: Double): Double = nativeMath.exp(x) Computes  $\exp(x) - 1$ . This function can be implemented to produce more precise result for  $x$  near zero. Special cases:  $\text{expm1}(\text{NaN})$  is  $\text{NaN}$ ;  $\text{expm1}(+\text{Inf})$  is  $+\text{Inf}$ ;  $\text{expm1}(-\text{Inf})$  is  $-1.0$ ; @see [exp] function.

Since Kotlin("1.2") Inline Only public actual inline fun expm1(x: Double): Double = nativeMath.expm1(x) Computes the logarithm of the value  $x$  to the given [base]. Special cases:  $\log(x, b)$  is  $\text{NaN}$  if either  $x$  or  $b$  are  $\text{NaN}$ ;  $\log(x, b)$  is  $\text{NaN}$  when  $x < 0$  or  $b \leq 0$  or  $b = 1.0$ ;  $\log(+\text{Inf}, +\text{Inf})$  is  $\text{NaN}$ ;  $\log(+\text{Inf}, b)$  is  $+\text{Inf}$  for  $b > 1$  and  $-\text{Inf}$  for  $b < 1$ ;  $\log(0.0, b)$  is  $-\text{Inf}$  for  $b > 1$  and  $+\text{Inf}$  for  $b < 1$ . See also logarithm functions for common fixed bases: [ln], [log10] and [log2].

Since Kotlin("1.2") public actual fun log(x: Double, base: Double): Double { if (base <= 0.0 || base == 1.0) return Double.NaN return nativeMath.log(x) / nativeMath.log(base)}

Computes the natural logarithm (base  $E$ ) of the value  $x$ .



Special cases:  $\ln(\text{NaN})$  is  $\text{NaN}$   $\ln(x)$  is  $\text{NaN}$  when  $x < 0.0$   $\ln(+\text{Inf})$  is  $+\text{Inf}$   $\ln(0.0)$  is  $-\text{Inf}$

```

@SinceKotlin("1.2")@InlineOnly@public actual inline fun ln(x: Double): Double
= nativeMath.log(x)
* Computes the common logarithm (base 10) of the value [x].
* @see [ln] function for special cases.
@SinceKotlin("1.2")@InlineOnly@public actual inline fun log10(x: Double): Double
= nativeMath.log10(x)
* Computes the binary logarithm (base 2) of the value [x].
* @see [ln] function for special cases.
@SinceKotlin("1.2")@InlineOnly@public actual inline fun log2(x: Double): Double
= nativeMath.log2(x)
* Computes $\ln(x + 1)$.
* This function can be implemented to produce more precise result for [x] near zero.
Special cases: $\ln1p(\text{NaN})$ is NaN $\ln1p(x)$ is NaN where $x < -1.0$ $\ln1p(-1.0)$ is $-\text{Inf}$ $\ln1p(+\text{Inf})$ is $+\text{Inf}$
* @see [ln] function
* @see [expm1] function
@SinceKotlin("1.2")@InlineOnly@public actual inline fun ln1p(x: Double): Double
= nativeMath.log1p(x)
* Rounds the given value [x] to an integer towards positive infinity.
* @return the smallest double value that is greater than or equal to the given value [x] and is a mathematical integer.
Special cases: $\text{ceil}(x)$ is x where x is NaN or $+\text{Inf}$ or $-\text{Inf}$ or already a mathematical integer.
@SinceKotlin("1.2")@InlineOnly@public actual inline fun ceil(x: Double): Double
= nativeMath.ceil(x).unsafeCast<Double>() // TODO: Remove unsafe cast after removing public js.math
* Rounds the given value [x] to an integer towards negative infinity.
* @return the largest double value that is smaller than or equal to the given value [x] and is a mathematical integer.
Special cases: $\text{floor}(x)$ is x where x is NaN or $+\text{Inf}$ or $-\text{Inf}$ or already a mathematical integer.
@SinceKotlin("1.2")@InlineOnly@public actual inline fun floor(x: Double): Double
= nativeMath.floor(x).unsafeCast<Double>()
* Rounds the given value [x] to an integer towards zero.
* @return the value [x] having its fractional part truncated.
Special cases: $\text{truncate}(x)$ is x where x is NaN or $+\text{Inf}$ or $-\text{Inf}$ or already a mathematical integer.
@SinceKotlin("1.2")@InlineOnly@public actual inline fun truncate(x: Double): Double
= nativeMath.trunc(x)
* Rounds the given value [x] towards the closest integer with ties rounded towards even integer.
Special cases: $\text{round}(x)$ is x where x is NaN or $+\text{Inf}$ or $-\text{Inf}$ or already a mathematical integer.
@SinceKotlin("1.2")@public actual fun round(x: Double): Double {
if (x % 0.5 != 0.0) {
return nativeMath.round(x).unsafeCast<Double>()
}
val floor = floor(x)
return if (floor % 2 == 0.0) floor else ceil(x)
}
* Returns the absolute value of the given value [x].
Special cases: $\text{abs}(\text{NaN})$ is NaN
* @see absoluteValue extension property for [Double]
@SinceKotlin("1.2")@InlineOnly@public actual inline fun abs(x: Double): Double
= nativeMath.abs(x)
* Returns the sign of the given value [x]: -1.0 if the value is negative, 0 if the value is zero, 1.0 if the value is positive
Special case: $\text{sign}(\text{NaN})$ is NaN
@SinceKotlin("1.2")@InlineOnly@public actual inline fun sign(x: Double): Double
= nativeMath.sign(x)
* Returns the smaller of two values.
* If either value is NaN , then the result is NaN .
@SinceKotlin("1.2")@InlineOnly@public actual inline fun min(a: Double, b: Double): Double
= nativeMath.min(a, b)
* Returns the greater of two values.
* If either value is NaN , then the result is NaN .
@SinceKotlin("1.2")@InlineOnly@public actual inline fun max(a: Double, b: Double): Double
= nativeMath.max(a, b)
// extensions
* Raises this value to the power [x].
Special cases: $b.\text{pow}(0.0)$ is 1.0 $b.\text{pow}(1.0) == b$ $b.\text{pow}(\text{NaN})$ is NaN $\text{NaN}.\text{pow}(x)$ is NaN for $x != 0.0$ $b.\text{pow}(\text{Inf})$ is NaN for $\text{abs}(b) == 1.0$ $b.\text{pow}(x)$ is NaN for $b < 0$ and x is finite and not an integer
@SinceKotlin("1.2")@InlineOnly@public actual inline fun Double.pow(x: Double): Double
= nativeMath.pow(this, x)
* Raises this value to the integer power [n].
* See the other overload of [pow] for details.
@SinceKotlin("1.2")@InlineOnly@public actual inline fun Double.pow(n: Int): Double
= nativeMath.pow(this, n.toDouble())
* Returns the absolute value of this value.
Special cases: $\text{NaN}.\text{absoluteValue}$ is NaN
* @see abs function
@SinceKotlin("1.2")@InlineOnly@public actual inline val Double.absoluteValue: Double get()
= nativeMath.abs(this)
* Returns the sign of this value: -1.0 if the value is negative, 0 if the value is zero, 1.0 if the value is positive

```

```

zero if the value is zero,\r\n * - `1.0` if the value is positive\r\n * \r\n * Special case:\r\n * - `NaN.sign` is
`NaN`\r\n * /\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline val Double.sign: Double get() =
nativeMath.sign(this)\r\n\r\n/**\r\n * Returns this value with the sign bit same as of the [sign] value.\r\n
*/\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun Double.withSign(sign: Int): Double =
this.withSign(sign.toDouble())\r\n\r\n/**\r\n * Returns the ulp (unit in the last place) of this value.\r\n * \r\n * An ulp
is a positive distance between this value and the next nearest [Double] value larger in magnitude.\r\n * \r\n * Special
Cases:\r\n * - `NaN.ulp` is `NaN`\r\n * - `x.ulp` is `+Inf` when `x` is `+Inf` or `-Inf`\r\n * - `0.0.ulp` is
`Double.MIN_VALUE`\r\n * /\r\n@SinceKotlin("1.2")\r\npublic actual val Double.ulp: Double get() = when {\r\n
this < 0 -> (-this).ulp\r\n this.isNaN() || this == Double.POSITIVE_INFINITY -> this\r\n this ==
Double.MAX_VALUE -> this - this.nextDown()\r\n else -> this.nextUp() - this\r\n}\r\n\r\n/**\r\n * Returns the
[Double] value nearest to this value in direction of positive infinity.\r\n * /\r\n@SinceKotlin("1.2")\r\npublic actual
fun Double.nextUp(): Double = when {\r\n this.isNaN() || this == Double.POSITIVE_INFINITY -> this\r\n this
== 0.0 -> Double.MIN_VALUE\r\n else -> Double.fromBits(this.toRawBits() + if (this > 0) 1 else -
1)\r\n}\r\n\r\n/**\r\n * Returns the [Double] value nearest to this value in direction of negative infinity.\r\n
*/\r\n@SinceKotlin("1.2")\r\npublic actual fun Double.nextDown(): Double = when {\r\n this.isNaN() || this ==
Double.NEGATIVE_INFINITY -> this\r\n this == 0.0 -> -Double.MIN_VALUE\r\n else ->
Double.fromBits(this.toRawBits() + if (this > 0) -1 else 1)\r\n}\r\n\r\n\r\n/**\r\n * Returns the [Double] value
nearest to this value in direction from this value towards the value [to].\r\n * \r\n * Special cases:\r\n * -
`x.nextTowards(y)` is `NaN` if either `x` or `y` are `NaN`\r\n * - `x.nextTowards(x) == x`\r\n * \r\n
*/\r\n@SinceKotlin("1.2")\r\npublic actual fun Double.nextTowards(to: Double): Double = when {\r\n
this.isNaN() || to.isNaN() -> Double.NaN\r\n to == this -> to\r\n to > this -> this.nextUp()\r\n else /* to < this
*/ -> this.nextDown()\r\n}\r\n\r\n\r\n/**\r\n * Rounds this [Double] value to the nearest integer and converts the
result to [Int].\r\n * \r\n * Ties are rounded towards positive infinity.\r\n * \r\n * Special cases:\r\n * - `x.roundToInt() ==
Int.MAX_VALUE` when `x > Int.MAX_VALUE`\r\n * - `x.roundToInt() == Int.MIN_VALUE` when `x <
Int.MIN_VALUE`\r\n * \r\n * @throws IllegalArgumentException when this value is `NaN`\r\n
*/\r\n@SinceKotlin("1.2")\r\npublic actual fun Double.roundToInt(): Int = when {\r\n isNaN() -> throw
IllegalArgumentException("Cannot round NaN value.")\r\n this > Int.MAX_VALUE -> Int.MAX_VALUE\r\n
this < Int.MIN_VALUE -> Int.MIN_VALUE\r\n else ->
nativeMath.round(this).unsafeCast<Double>().toInt()\r\n}\r\n\r\n\r\n/**\r\n * Rounds this [Double] value to the nearest
integer and converts the result to [Long].\r\n * \r\n * Ties are rounded towards positive infinity.\r\n * \r\n * Special
cases:\r\n * - `x.roundToLong() == Long.MAX_VALUE` when `x > Long.MAX_VALUE`\r\n * -
`x.roundToLong() == Long.MIN_VALUE` when `x < Long.MIN_VALUE`\r\n * \r\n * @throws
IllegalArgumentException when this value is `NaN`\r\n * /\r\n@SinceKotlin("1.2")\r\npublic actual fun
Double.roundToLong(): Long = when {\r\n isNaN() -> throw IllegalArgumentException("Cannot round NaN
value.")\r\n this > Long.MAX_VALUE -> Long.MAX_VALUE\r\n this < Long.MIN_VALUE ->
Long.MIN_VALUE\r\n else -> nativeMath.round(this).unsafeCast<Double>().toLong()\r\n}\r\n\r\n\r\n//
endregion\r\n\r\n\r\n\r\n// region ===== Float Math
=====
\r\n\r\n\r\n/**\r\n * Computes the sine of the angle [x] given in
radians.\r\n * \r\n * Special cases:\r\n * - `sin(NaN|+Inf|-Inf)` is `NaN`\r\n
*/\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun sin(x: Float): Float =
nativeMath.sin(x.toDouble()).toFloat()\r\n\r\n\r\n/**\r\n * Computes the cosine of the angle [x] given in radians.\r\n
*/\r\n * \r\n * Special cases:\r\n * - `cos(NaN|+Inf|-Inf)` is `NaN`\r\n * /\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic
actual inline fun cos(x: Float): Float = nativeMath.cos(x.toDouble()).toFloat()\r\n\r\n\r\n/**\r\n * Computes the tangent of the
angle [x] given in radians.\r\n * \r\n * Special cases:\r\n * - `tan(NaN|+Inf|-Inf)` is `NaN`\r\n
*/\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun tan(x: Float): Float =
nativeMath.tan(x.toDouble()).toFloat()\r\n\r\n\r\n/**\r\n * Computes the arc sine of the value [x];\r\n * \r\n * the returned
value is an angle in the range from $-\pi/2$ to $\pi/2$ radians.\r\n * \r\n * Special cases:\r\n * - `asin(x)` is `NaN`,
when `abs(x) > 1` or x is `NaN`\r\n * /\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun asin(x:

```

Float): Float = nativeMath.asin(x.toDouble()).toFloat()  
 \* Computes the arc cosine of the value [x];  
 \* the returned value is an angle in the range from  $0.0$  to  $\pi$  radians.  
 \* Special cases:  
 \*  $\text{acos}(x)$  is  $\text{NaN}$ , when  $\text{abs}(x) > 1$  or  $x$  is  $\text{NaN}$   
 @SinceKotlin("1.2")@InlineOnly\npublic actual inline fun  
 acos(x: Float): Float = nativeMath.acos(x.toDouble()).toFloat()  
 \* Computes the arc tangent of the  
 value [x];  
 \* the returned value is an angle in the range from  $-\pi/2$  to  $\pi/2$  radians.  
 \* Special cases:  
 \*  $\text{atan}(\text{NaN})$  is  $\text{NaN}$   
 @SinceKotlin("1.2")@InlineOnly\npublic actual inline fun  
 atan(x: Float):  
 Float = nativeMath.atan(x.toDouble()).toFloat()  
 \* Returns the angle  $\theta$  of the polar coordinates  $(r, \theta)$   
 that correspond to the rectangular coordinates  $(x, y)$  by computing the arc tangent of the value  $y /$   
 $x$ ;  
 \* the returned value is an angle in the range from  $-\pi$  to  $\pi$  radians.  
 \* Special cases:  
 \*  $\text{atan2}(0.0, 0.0)$  is  $0.0$   
 \*  $\text{atan2}(0.0, x)$  is  $0.0$  for  $x > 0$  and  $\pi$  for  $x < 0$   
 \*  $\text{atan2}(-0.0, x)$  is  $-0.0$  for  $x > 0$  and  $-\pi$  for  $x < 0$   
 \*  $\text{atan2}(y, +\text{Inf})$  is  $0.0$  for  $0 < y < +\text{Inf}$  and  $-0.0$  for  $-\text{Inf} < y < 0$   
 \*  $\text{atan2}(y, -\text{Inf})$  is  $\pi$  for  $0 < y < +\text{Inf}$  and  $-\pi$  for  $-\text{Inf} < y < 0$   
 \*  $\text{atan2}(y, 0.0)$  is  $\pi/2$  for  $y > 0$   
 and  $-\pi/2$  for  $y < 0$   
 \*  $\text{atan2}(+\text{Inf}, x)$  is  $\pi/2$  for finite  $x$   
 \*  $\text{atan2}(-\text{Inf}, x)$  is  $-\pi/2$  for finite  
 $x$   
 \*  $\text{atan2}(\text{NaN}, x)$  and  $\text{atan2}(y, \text{NaN})$  is  $\text{NaN}$   
 @SinceKotlin("1.2")@InlineOnly\npublic  
 actual inline fun atan2(y: Float, x: Float): Float = nativeMath.atan2(y.toDouble(),  
 x.toDouble()).toFloat()  
 \* Computes the hyperbolic sine of the value [x].  
 \* Special cases:  
 \*  $\text{sinh}(\text{NaN})$  is  $\text{NaN}$   
 \*  $\text{sinh}(+\text{Inf})$  is  $+\text{Inf}$   
 \*  $\text{sinh}(-\text{Inf})$  is  $-\text{Inf}$   
 @SinceKotlin("1.2")@InlineOnly\npublic actual inline fun  
 sinh(x: Float): Float =  
 nativeMath.sinh(x.toDouble()).toFloat()  
 \* Computes the hyperbolic cosine of the value [x].  
 \* Special cases:  
 \*  $\text{cosh}(\text{NaN})$  is  $\text{NaN}$   
 \*  $\text{cosh}(+\text{Inf}-\text{Inf})$  is  $+\text{Inf}$   
 @SinceKotlin("1.2")@InlineOnly\npublic actual inline fun  
 cosh(x: Float): Float =  
 nativeMath.cosh(x.toDouble()).toFloat()  
 \* Computes the hyperbolic tangent of the value [x].  
 \* Special cases:  
 \*  $\text{tanh}(\text{NaN})$  is  $\text{NaN}$   
 \*  $\text{tanh}(+\text{Inf})$  is  $1.0$   
 \*  $\text{tanh}(-\text{Inf})$  is  $-1.0$   
 @SinceKotlin("1.2")@InlineOnly\npublic actual inline fun  
 tanh(x: Float): Float =  
 nativeMath.tanh(x.toDouble()).toFloat()  
 \* Computes the inverse hyperbolic sine of the value [x].  
 \* The returned value is  $y$  such that  $\text{sinh}(y) == x$ .  
 \* Special cases:  
 \*  $\text{asinh}(\text{NaN})$  is  $\text{NaN}$   
 \*  $\text{asinh}(+\text{Inf})$  is  $+\text{Inf}$   
 \*  $\text{asinh}(-\text{Inf})$  is  $-\text{Inf}$   
 @SinceKotlin("1.2")@InlineOnly\npublic  
 actual inline fun asinh(x: Float): Float = nativeMath.asinh(x.toDouble()).toFloat()  
 \* Computes the  
 inverse hyperbolic cosine of the value [x].  
 \* The returned value is positive  $y$  such that  $\text{cosh}(y) == x$ .  
 \* Special cases:  
 \*  $\text{acosh}(\text{NaN})$  is  $\text{NaN}$   
 \*  $\text{acosh}(x)$  is  $\text{NaN}$  when  $x < 1$   
 \*  $\text{acosh}(+\text{Inf})$  is  $+\text{Inf}$   
 @SinceKotlin("1.2")@InlineOnly\npublic actual inline fun  
 acosh(x: Float): Float =  
 nativeMath.acosh(x.toDouble()).toFloat()  
 \* Computes the inverse hyperbolic tangent of the value  
 $[x]$ .  
 \* The returned value is  $y$  such that  $\text{tanh}(y) == x$ .  
 \* Special cases:  
 \*  $\text{atanh}(\text{NaN})$  is  
 $\text{NaN}$   
 \*  $\text{atanh}(x)$  is  $\text{NaN}$  when  $x > 1$  or  $x < -1$   
 \*  $\text{atanh}(1.0)$  is  $+\text{Inf}$   
 \*  $\text{atanh}(-1.0)$  is  $-\text{Inf}$   
 @SinceKotlin("1.2")@InlineOnly\npublic actual inline fun  
 atanh(x: Float): Float =  
 nativeMath.atanh(x.toDouble()).toFloat()  
 \* Computes  $\sqrt{x^2 + y^2}$  without intermediate overflow  
 or underflow.  
 \* Special cases:  
 \* returns  $+\text{Inf}$  if any of arguments is infinite  
 \* returns  $\text{NaN}$  if  
 any of arguments is  $\text{NaN}$  and the other is not infinite  
 @SinceKotlin("1.2")@InlineOnly\npublic  
 actual inline fun hypot(x: Float, y: Float): Float = nativeMath.hypot(x.toDouble(),  
 y.toDouble()).toFloat()  
 \* Computes the positive square root of the value [x].  
 \* Special  
 cases:  
 \*  $\sqrt{x}$  is  $\text{NaN}$  when  $x < 0$  or  $x$  is  $\text{NaN}$   
 @SinceKotlin("1.2")@InlineOnly\npublic actual inline fun  
 sqrt(x: Float): Float =  
 nativeMath.sqrt(x.toDouble()).toFloat()  
 \* Computes Euler's number  $e$  raised to the power of the  
 value [x].  
 \* Special cases:  
 \*  $\text{exp}(\text{NaN})$  is  $\text{NaN}$   
 \*  $\text{exp}(+\text{Inf})$  is  $+\text{Inf}$   
 \*  $\text{exp}(-\text{Inf})$  is  
 $0.0$   
 @SinceKotlin("1.2")@InlineOnly\npublic actual inline fun  
 exp(x: Float): Float =  
 nativeMath.exp(x.toDouble()).toFloat()  
 \* Computes  $\exp(x) - 1$ .  
 \* This function can be  
 implemented to produce more precise result for [x] near zero.  
 \* Special cases:  
 \*  $\text{expm1}(\text{NaN})$  is  
 $\text{NaN}$   
 \*  $\text{expm1}(+\text{Inf})$  is  $+\text{Inf}$   
 \*  $\text{expm1}(-\text{Inf})$  is  $-1.0$   
 \* @see [exp] function.

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun expm1(x: Float): Float = nativeMath.expm1(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Computes the logarithm of the value [x] to the given [base].\r\n * Special cases:\r\n * -  $\log(x, b)$  is  $\text{NaN}$  if either  $x$  or  $b$  are  $\text{NaN}$ \r\n * -  $\log(x, b)$  is  $\text{NaN}$  when  $x < 0$  or  $b \leq 0$  or  $b == 1.0$ \r\n * -  $\log(+\text{Inf}, +\text{Inf})$  is  $\text{NaN}$ \r\n * -  $\log(+\text{Inf}, b)$  is  $+\text{Inf}$  for  $b > 1$  and  $-\text{Inf}$  for  $b < 1$ \r\n * -  $\log(0.0, b)$  is  $-\text{Inf}$  for  $b > 1$  and  $+\text{Inf}$  for  $b < 1$ \r\n * See also logarithm functions for common fixed bases: [ln], [log10] and [log2].\r\n`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun log(x: Float, base: Float): Float = log(x.toDouble(), base.toDouble()).toFloat()\r\n\r\n/**\r\n * Computes the natural logarithm (base  $E$ ) of the value [x].\r\n * Special cases:\r\n * -  $\ln(\text{NaN})$  is  $\text{NaN}$ \r\n * -  $\ln(x)$  is  $\text{NaN}$  when  $x < 0.0$ \r\n * -  $\ln(+\text{Inf})$  is  $+\text{Inf}$ \r\n * -  $\ln(0.0)$  is  $-\text{Inf}$ \r\n`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun ln(x: Float): Float = nativeMath.log(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Computes the common logarithm (base 10) of the value [x].\r\n * @see [ln] function for special cases.\r\n`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun log10(x: Float): Float = nativeMath.log10(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Computes the binary logarithm (base 2) of the value [x].\r\n * @see [ln] function for special cases.\r\n`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun log2(x: Float): Float = nativeMath.log2(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Computes  $\ln(a + 1)$ .  
 * This function can be implemented to produce more precise result for [x] near zero.  
 * Special cases:  
 * -  $\ln1p(\text{NaN})$  is  $\text{NaN}$   
 * -  $\ln1p(x)$  is  $\text{NaN}$  where  $x < -1.0$   
 * -  $\ln1p(-1.0)$  is  $-\text{Inf}$   
 * -  $\ln1p(+\text{Inf})$  is  $+\text{Inf}$   
 * @see [ln] function  
 * @see [expm1] function\r\n`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun ln1p(x: Float): Float = nativeMath.log1p(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Rounds the given value [x] to an integer towards positive infinity.  
 * @return the smallest Float value that is greater than or equal to the given value [x] and is a mathematical integer.  
 * Special cases:  
 * -  $\text{ceil}(x)$  is  $x$  where  $x$  is  $\text{NaN}$  or  $+\text{Inf}$  or  $-\text{Inf}$  or already a mathematical integer.`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun ceil(x: Float): Float = nativeMath.ceil(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Rounds the given value [x] to an integer towards negative infinity.  
 * @return the largest Float value that is smaller than or equal to the given value [x] and is a mathematical integer.  
 * Special cases:  
 * -  $\text{floor}(x)$  is  $x$  where  $x$  is  $\text{NaN}$  or  $+\text{Inf}$  or  $-\text{Inf}$  or already a mathematical integer.`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun floor(x: Float): Float = nativeMath.floor(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Rounds the given value [x] to an integer towards zero.  
 * @return the value [x] having its fractional part truncated.  
 * Special cases:  
 * -  $\text{truncate}(x)$  is  $x$  where  $x$  is  $\text{NaN}$  or  $+\text{Inf}$  or  $-\text{Inf}$  or already a mathematical integer.`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun truncate(x: Float): Float = truncate(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Rounds the given value [x] towards the closest integer with ties rounded towards even integer.  
 * Special cases:  
 * -  $\text{round}(x)$  is  $x$  where  $x$  is  $\text{NaN}$  or  $+\text{Inf}$  or  $-\text{Inf}$  or already a mathematical integer.`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun round(x: Float): Float = round(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Returns the absolute value of the given value [x].  
 * Special cases:  
 * -  $\text{abs}(\text{NaN})$  is  $\text{NaN}$   
 * @see absoluteValue extension property for [Float]\r\n`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun abs(x: Float): Float = nativeMath.abs(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Returns the sign of the given value [x]:  
 * -  $-1.0$  if the value is negative,  
 * - zero if the value is zero,  
 * -  $1.0$  if the value is positive\r\n`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun sign(x: Float): Float = nativeMath.sign(x.toDouble()).toFloat()\r\n\r\n/**\r\n * Returns the smaller of two values.  
 * If either value is  $\text{NaN}$ , then the result is  $\text{NaN}$ .`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun min(a: Float, b: Float): Float = nativeMath.min(a, b)\r\n\r\n/**\r\n * Returns the greater of two values.  
 * If either value is  $\text{NaN}$ , then the result is  $\text{NaN}$ .`

`*\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\npublic actual inline fun max(a: Float, b: Float): Float = nativeMath.max(a, b)\r\n\r\n// extensions\r\n\r\n/**\r\n * Raises this value to the power [x].  
 * Special cases:  
 * -  $b.\text{pow}(0.0)$  is  $1.0$   
 * -  $b.\text{pow}(1.0) == b$   
 * -  $b.\text{pow}(\text{NaN})$  is  $\text{NaN}$   
 * -  $\text{NaN}.\text{pow}(x)$  is  $\text{NaN}$  for  $x != 0.0$   
 * -  $b.\text{pow}(\text{Inf})$  is  $\text{NaN}$  for  $\text{abs}(b) ==$`

`1.0` \* - `b.pow(x)` is `NaN` for `b < 0` and `x` is finite and not an integer  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline fun Float.pow(x: Float): Float =`  
`nativeMath.pow(this.toDouble(), x.toDouble()).toFloat()` \* Raises this value to the integer power  
`[n]`. \* See the other overload of `[pow]` for details.  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline fun Float.pow(n: Int): Float =`  
`nativeMath.pow(this.toDouble(), n.toDouble()).toFloat()` \* Returns the absolute value of this value.  
 \* Special cases: \* - `NaN.absoluteValue` is `NaN` \*  
`@see abs function`  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline val Float.absoluteValue: Float`  
`get() = nativeMath.abs(this.toDouble()).toFloat()` \* Returns the sign of this value:  
 \* - `-1.0` if the value is negative,  
 \* - zero if the value is zero,  
 \* - `1.0` if the value is positive \* Special case:  
 \* - `NaN.sign` is `NaN`  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline val Float.sign: Float`  
`get() = nativeMath.sign(this.toDouble()).toFloat()` \* Returns this value with the sign  
 bit same as of the `[sign]` value. \* If `[sign]` is `NaN` the sign of the result is undefined.  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline fun Float.withSign(sign: Float): Float =`  
`this.toDouble().withSign(sign.toDouble()).toFloat()` \* Returns this value with the sign bit same as of  
 the `[sign]` value.  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline fun Float.withSign(sign: Int): Float =`  
`this.toDouble().withSign(sign.toDouble()).toFloat()` \* Rounds this `[Float]` value to the  
 nearest integer and converts the result to `[Int]`. \* Ties are rounded towards positive infinity.  
 \* Special cases: \* - `x.roundToInt() == Int.MAX_VALUE` when `x > Int.MAX_VALUE` \* - `x.roundToInt() ==`  
`Int.MIN_VALUE` when `x < Int.MIN_VALUE` \* @throws `IllegalArgumentException` when this value is  
`NaN`  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline fun Float.roundToInt(): Int =`  
`toDouble().roundToInt()` \* Rounds this `[Float]` value to the nearest integer and converts the result to  
`[Long]`. \* Ties are rounded towards positive infinity. \* Special cases: \* - `x.roundToLong() ==`  
`Long.MAX_VALUE` when `x > Long.MAX_VALUE` \* - `x.roundToLong() == Long.MIN_VALUE` when  
`x < Long.MIN_VALUE` \* @throws `IllegalArgumentException` when this value is `NaN`  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline fun Float.roundToLong(): Long =`  
`toDouble().roundToLong()` // `endregion` // `region` ===== `Integer Math`  
 =====  
 \* Returns the absolute value of the given  
 value `[n]`. \* Special cases: \* - `abs(Int.MIN_VALUE)` is `Int.MIN_VALUE` due to an overflow  
 \* @see `absoluteValue` extension property for `[Int]`  
`@SinceKotlin("1.2")` `public actual fun abs(n: Int): Int =`  
`if (n < 0) (-n or 0) else n` \* Returns the smaller of two values.  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline fun min(a: Int, b: Int): Int =`  
`nativeMath.min(a, b)` \* Returns the greater of two values.  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline fun max(a: Int, b: Int): Int =`  
`nativeMath.max(a, b)` \* Returns the absolute value of this value. \* Special cases: \* -  
`Int.MIN_VALUE.absoluteValue` is `Int.MIN_VALUE` due to an overflow \* @see `abs function`  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline val Int.absoluteValue: Int`  
`get() = abs(this)` \* Returns the sign of this value: \* - `-1` if the value is negative,  
 \* - `0` if the value is zero,  
 \* - `1` if the value is positive  
`@SinceKotlin("1.2")` `public actual val Int.sign: Int`  
`get() = when {`  
`this < 0 -> -1`  
`this > 0 -> 1`  
`else -> 0`  
`}` \* Returns the absolute value of the  
 given value `[n]`. \* Special cases: \* - `abs(Long.MIN_VALUE)` is `Long.MIN_VALUE` due to an  
 overflow \* @see `absoluteValue` extension property for `[Long]`  
`@SinceKotlin("1.2")` `public actual fun abs(n: Long): Long =`  
`if (n < 0) -n else n` \* Returns the smaller of two values.  
`@SinceKotlin("1.2")` `Suppress("NOTHING_TO_INLINE")` `public actual inline fun min(a: Long, b: Long): Long =`  
`if (a <= b) a else b` \* Returns the greater of two values.  
`@SinceKotlin("1.2")` `Suppress("NOTHING_TO_INLINE")` `public actual inline fun max(a: Long, b: Long): Long =`  
`if (a >= b) a else b` \* Returns the absolute value of this value. \* Special cases:  
 \* - `Long.MIN_VALUE.absoluteValue` is `Long.MIN_VALUE` due to an overflow \* @see  
`abs function`  
`@SinceKotlin("1.2")` `InlineOnly` `public actual inline val Long.absoluteValue: Long`

```

get() = abs(this)\r\n\r\n/**\r\n * Returns the sign of this value:\r\n * - `1` if the value is negative,\r\n * - `0` if the
value is zero,\r\n * - `1` if the value is positive\r\n */\r\n@SinceKotlin("1.2")\r\npublic actual val Long.sign: Int
get() = when {\r\n this < 0 -> -1\r\n this > 0 -> 1\r\n else -> 0\r\n}\r\n\r\n\r\n// endregion\r\n"/**\r\n *
Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code
is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage
kotlin\r\n\r\n/**\r\n * Returns `true` if the specified number is a\r\n * Not-a-Number (NaN) value, `false`
otherwise.\r\n */\r\npublic actual fun Double.isNaN(): Boolean = this != this\r\n\r\n/**\r\n * Returns `true` if the
specified number is a\r\n * Not-a-Number (NaN) value, `false` otherwise.\r\n */\r\npublic actual fun Float.isNaN():
Boolean = this != this\r\n\r\n/**\r\n * Returns `true` if this value is infinitely large in magnitude.\r\n */\r\npublic
actual fun Double.isInfinite(): Boolean = this == Double.POSITIVE_INFINITY || this ==
Double.NEGATIVE_INFINITY\r\n\r\n/**\r\n * Returns `true` if this value is infinitely large in magnitude.\r\n
*/\r\npublic actual fun Float.isInfinite(): Boolean = this == Float.POSITIVE_INFINITY || this ==
Float.NEGATIVE_INFINITY\r\n\r\n/**\r\n * Returns `true` if the argument is a finite floating-point value; returns
`false` otherwise (for `NaN` and infinity arguments).\r\n */\r\npublic actual fun Double.isFinite(): Boolean =
!isInfinite() && !isNaN()\r\n\r\n/**\r\n * Returns `true` if the argument is a finite floating-point value; returns
`false` otherwise (for `NaN` and infinity arguments).\r\n */\r\npublic actual fun Float.isFinite(): Boolean =
!isInfinite() && !isNaN()\r\n\r\n/**\r\n * Counts the number of set bits in the binary representation of this [Int]
number.\r\n */\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun Int.countOneBits(): Int
{\r\n // Hacker's Delight 5-1 algorithm\r\n var v = this\r\n v = (v and 0x55555555) + (v.ushr(1) and
0x55555555)\r\n v = (v and 0x33333333) + (v.ushr(2) and 0x33333333)\r\n v = (v and 0x0F0F0F0F) +
(v.ushr(4) and 0x0F0F0F0F)\r\n v = (v and 0x00FF00FF) + (v.ushr(8) and 0x00FF00FF)\r\n v = (v and
0x0000FFFF) + (v.ushr(16))\r\n return v\r\n}\r\n\r\n/**\r\n * Counts the number of consecutive most significant
bits that are zero in the binary representation of this [Int] number.\r\n
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@Suppress("DEPRECATION")\r\n@kotlin.internal.In
lineOnly\r\npublic actual inline fun Int.countLeadingZeroBits(): Int = kotlin.js.Math.clz32(this)\r\n\r\n/**\r\n *
Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int]
number.\r\n */\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun
Int.countTrailingZeroBits(): Int =\r\n // Hacker's Delight 5-4 algorithm for expressing countTrailingZeroBits with
countLeadingZeroBits\r\n Int.SIZE_BITS - (this or -this).inv().countLeadingZeroBits()\r\n\r\n/**\r\n * Returns a
number having a single bit set in the position of the most significant set bit of this [Int] number,\r\n * or zero, if this
number is zero.\r\n */\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun
Int.takeHighestOneBit(): Int =\r\n if (this == 0) 0 else 1.shl(Int.SIZE_BITS - 1 -
countLeadingZeroBits())\r\n\r\n/**\r\n * Returns a number having a single bit set in the position of the least
significant set bit of this [Int] number,\r\n * or zero, if this number is zero.\r\n
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun Int.takeLowestOneBit(): Int =\r\n //
Hacker's Delight 2-1 algorithm for isolating rightmost 1-bit\r\n this and -this\r\n\r\n/**\r\n * Rotates the binary
representation of this [Int] number left by the specified [bitCount] number of bits.\r\n * The most significant bits
pushed out from the left side reenter the number as the least significant bits on the right side.\r\n */\r\n * Rotating the
number left by a negative bit count is the same as rotating it right by the negated bit count:\r\n *
`number.rotateLeft(-n) == number.rotateRight(n)`\r\n\r\n * Rotating by a multiple of [Int.SIZE_BITS] (32) returns
the same number, or more generally\r\n * `number.rotateLeft(n) == number.rotateLeft(n % 32)`\r\n
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun Int.rotateLeft(bitCount: Int): Int =\r\n shl(bitCount) or ushr(Int.SIZE_BITS - bitCount)\r\n\r\n/**\r\n * Rotates the binary representation of this [Int]
number right by the specified [bitCount] number of bits.\r\n * The least significant bits pushed out from the right
side reenter the number as the most significant bits on the left side.\r\n */\r\n * Rotating the number right by a
negative bit count is the same as rotating it left by the negated bit count:\r\n *
`number.rotateRight(-n) ==
number.rotateLeft(n)`\r\n\r\n * Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more
generally\r\n * `number.rotateRight(n) == number.rotateRight(n % 32)`\r\n

```

```

\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun Int.rotateRight(bitCount: Int): Int =\r\n shl(Int.SIZE_BITS - bitCount) or ushr(bitCount)\r\n\r\n\r\n\r\n\r\n * Counts the number of set bits in the binary\r\n representation of this [Long] number.\r\n\r\n*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual\r\n fun Long.countOneBits(): Int =\r\n high.countOneBits() + low.countOneBits()\r\n\r\n\r\n\r\n*\r\n * Counts the number\r\n of consecutive most significant bits that are zero in the binary representation of this [Long] number.\r\n\r\n*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun Long.countLeadingZeroBits(): Int\r\n =\r\n when (val high = this.high) {\r\n 0 -> Int.SIZE_BITS + low.countLeadingZeroBits()\r\n else ->\r\n high.countLeadingZeroBits()\r\n }\r\n\r\n\r\n\r\n*\r\n * Counts the number of consecutive least significant bits that are\r\n zero in the binary representation of this [Long] number.\r\n\r\n*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun Long.countTrailingZeroBits(): Int\r\n =\r\n when (val low = this.low) {\r\n 0 -> Int.SIZE_BITS + high.countTrailingZeroBits()\r\n else ->\r\n low.countTrailingZeroBits()\r\n }\r\n\r\n\r\n\r\n*\r\n * Returns a number having a single bit set in the position of the\r\n most significant set bit of this [Long] number, or zero, if this number is zero.\r\n\r\n*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun Long.takeHighestOneBit(): Long =\r\n when (val high = this.high) {\r\n 0 -> Long(low.takeHighestOneBit(), 0)\r\n else -> Long(0,\r\n high.takeHighestOneBit())\r\n }\r\n\r\n\r\n\r\n*\r\n * Returns a number having a single bit set in the position of the least\r\n significant set bit of this [Long] number, or zero, if this number is zero.\r\n\r\n*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun Long.takeLowestOneBit(): Long =\r\n when (val low = this.low) {\r\n 0 -> Long(0, high.takeLowestOneBit())\r\n else ->\r\n Long(low.takeLowestOneBit(), 0)\r\n }\r\n\r\n\r\n\r\n*\r\n * Rotates the binary representation of this [Long] number\r\n left by the specified [bitCount] number of bits. The most significant bits pushed out from the left side reenter\r\n the number as the least significant bits on the right side. Rotating the number left by a negative bit count\r\n is the same as rotating it right by the negated bit count: `number.rotateLeft(-n) == number.rotateRight(n)`\r\n*\r\n * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\r\n * `number.rotateLeft(n) == number.rotateLeft(n % 64)`\r\n\r\n*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic actual fun Long.rotateLeft(bitCount: Int): Long\r\n {\r\n if ((bitCount and 31) != 0) {\r\n val low = this.low\r\n val high = this.high\r\n val newLow =\r\n low.shl(bitCount) or high.ushr(-bitCount)\r\n val newHigh = high.shl(bitCount) or low.ushr(-bitCount)\r\n return if ((bitCount and 32) == 0) Long(newLow, newHigh) else Long(newHigh, newLow)\r\n } else {\r\n return if ((bitCount and 32) == 0) this else Long(high, low)\r\n }\r\n}\r\n\r\n\r\n\r\n*\r\n * Rotates the binary\r\n representation of this [Long] number right by the specified [bitCount] number of bits. The least significant bits\r\n pushed out from the right side reenter the number as the most significant bits on the left side. Rotating the\r\n number right by a negative bit count is the same as rotating it left by the negated bit count:\r\n * `number.rotateRight(-n) == number.rotateLeft(n)`\r\n*\r\n * Rotating by a multiple of [Long.SIZE_BITS] (64)\r\n returns the same number, or more generally\r\n * `number.rotateRight(n) == number.rotateRight(n % 64)`\r\n\r\n*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@kotlin.internal.InlineOnly\r\npublic actual inline fun\r\n Long.rotateRight(bitCount: Int): Long = rotateLeft(-bitCount)\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o.\r\n and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license\r\n that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.js\r\n\r\nimport\r\n kotlin.internal.LowPriorityInOverloadResolution\r\n\r\n\r\n\r\n*\r\n * Exposes the JavaScript [Promise\r\n object](https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Global_Objects/Promise) to Kotlin.\r\n\r\n*\r\n@Suppress("NOT_DOCUMENTED")\r\npublic open external class Promise<out T>(executor: (resolve: (T)\r\n -> Unit, reject: (Throwable) -> Unit) -> Unit) {\r\n @LowPriorityInOverloadResolution\r\n public open fun <S>\r\n then(onFulfilled: ((T) -> S)?): Promise<S>\r\n\r\n @LowPriorityInOverloadResolution\r\n public open fun <S>\r\n then(onFulfilled: ((T) -> S)?, onRejected: ((Throwable) -> S)?): Promise<S>\r\n\r\n public open fun <S>\r\n catch(onRejected: (Throwable) -> S): Promise<S>\r\n\r\n companion object {\r\n public fun <S> all(promise:\r\n Array<out Promise<S>>): Promise<Array<out S>>\r\n\r\n public fun <S> race(promise: Array<out\r\n Promise<S>>): Promise<S>\r\n\r\n public fun reject(e: Throwable): Promise<Nothing>\r\n\r\n public fun

```





```

Any?): Boolean {
 if (other !is PrimitiveKClassImpl<*>) return false
 return super.equals(other) &&
 givenSimpleName == other.givenSimpleName
}
override val simpleName: String? get() =
 givenSimpleName
override fun isInstance(value: Any?): Boolean {
 return
 isInstanceFunction(value)
}
internal object NothingKClassImpl :
 KClassImpl<Nothing>(js("Object")) {
 override val simpleName: String = "Nothing"
 override fun
 isInstance(value: Any?): Boolean = false
 override val jClass: JsClass<Nothing>
 get() = throw
 UnsupportedOperationException("There's no native JS class for Nothing type")
 override fun equals(other:
 Any?): Boolean = other === this
 override fun hashCode(): Int = 0
}
/*
 * Copyright 2010-2018
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the
 * Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package
 kotlin.reflect.js.internal
import kotlin.js.JsClass
@JsName("PrimitiveClasses")
internal object
 PrimitiveClasses {
 @JsName("anyClass")
 val anyClass =
 PrimitiveKClassImpl(js("Object").unsafeCast<JsClass<Any>>(), "Any", { it is Any })
 @JsName("numberClass")
 val numberClass =
 PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Number>>(), "Number", { it is Number })
 @JsName("nothingClass")
 val nothingClass = NothingKClassImpl
 @JsName("booleanClass")
 val booleanClass = PrimitiveKClassImpl(js("Boolean").unsafeCast<JsClass<Boolean>>(), "Boolean", { it is
 Boolean })
 @JsName("byteClass")
 val byteClass =
 PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Byte>>(), "Byte", { it is Byte })
 @JsName("shortClass")
 val shortClass =
 PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Short>>(), "Short", { it is Short })
 @JsName("intClass")
 val intClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Int>>(),
 "Int", { it is Int })
 @JsName("floatClass")
 val floatClass =
 PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Float>>(), "Float", { it is Float })
 @JsName("doubleClass")
 val doubleClass =
 PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Double>>(), "Double", { it is Double })
 @JsName("arrayClass")
 val arrayClass =
 PrimitiveKClassImpl(js("Array").unsafeCast<JsClass<Array<*>>(), "Array", { it is Array<*> })
 @JsName("stringClass")
 val stringClass =
 PrimitiveKClassImpl(js("String").unsafeCast<JsClass<String>>(), "String", { it is String })
 @JsName("throwableClass")
 val throwableClass =
 PrimitiveKClassImpl(js("Error").unsafeCast<JsClass<Throwable>>(), "Throwable", { it is Throwable })
 @JsName("booleanArrayClass")
 val booleanArrayClass =
 PrimitiveKClassImpl(js("Array").unsafeCast<JsClass<BooleanArray>>(), "BooleanArray", { it is BooleanArray
 })
 @JsName("charArrayClass")
 val charArrayClass =
 PrimitiveKClassImpl(js("Uint16Array").unsafeCast<JsClass<CharArray>>(), "CharArray", { it is CharArray
 })
 @JsName("byteArrayClass")
 val byteArrayClass =
 PrimitiveKClassImpl(js("Int8Array").unsafeCast<JsClass<ByteArray>>(), "ByteArray", { it is ByteArray
 })
 @JsName("shortArrayClass")
 val shortArrayClass =
 PrimitiveKClassImpl(js("Int16Array").unsafeCast<JsClass<ShortArray>>(), "ShortArray", { it is ShortArray
 })
 @JsName("intArrayClass")
 val intArrayClass =
 PrimitiveKClassImpl(js("Int32Array").unsafeCast<JsClass<IntArray>>(), "IntArray", { it is IntArray })
 @JsName("longArrayClass")
 val longArrayClass =
 PrimitiveKClassImpl(js("Array").unsafeCast<JsClass<LongArray>>(), "LongArray", { it is LongArray })
 @JsName("floatArrayClass")
 val floatArrayClass =
 PrimitiveKClassImpl(js("Float32Array").unsafeCast<JsClass<FloatArray>>(), "FloatArray", { it is FloatArray
 })
 @JsName("doubleArrayClass")
 val doubleArrayClass =
 PrimitiveKClassImpl(js("Float64Array").unsafeCast<JsClass<DoubleArray>>(), "DoubleArray", { it is

```

```

DoubleArray })\r\n\r\n @JsName("functionClass")\r\n fun functionClass(arity: Int): KClassImpl<Any> {\r\n
return functionClasses.get(arity) ?: run {\r\n val result =
PrimitiveKClassImpl(js("Function").unsafeCast<JsClass<Any>>(), "Function$arity"),\r\n
{ jsTypeOf(it) === "function" && it.asDynamic().length === arity }}\r\n
functionClasses.asDynamic()[arity] = result\r\n result\r\n }\r\n }\r\n\r\nprivate val functionClasses
= arrayOfNulls<KClassImpl<Any>>(0), "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\n// a package is omitted to get declarations directly under the
module\r\n\r\nimport kotlin.reflect.KClass\r\nimport
kotlin.reflect.js.internal.*\r\n\r\n@JsName("getKClass")\r\ninternal fun <T : Any> getKClass(jClass: JsClass<T>):
KClass<T> = getOrCreateKClass(jClass)\r\n\r\n@JsName("getKClassFromExpression")\r\ninternal fun <T : Any>
getKClassFromExpression(e: T): KClass<T> =\r\n when (jsTypeOf(e)) {\r\n "string" ->
PrimitiveClasses.stringClass\r\n "number" -> if (jsBitwiseOr(e, 0).asDynamic() === e)
PrimitiveClasses.intClass else PrimitiveClasses.doubleClass\r\n "boolean" ->
PrimitiveClasses.booleanClass\r\n "function" -> PrimitiveClasses.functionClass(e.asDynamic().length)\r\n
else -> {\r\n when {\r\n e is BooleanArray -> PrimitiveClasses.booleanArrayClass\r\n
CharArray -> PrimitiveClasses.charArrayClass\r\n e is ByteArray -> PrimitiveClasses.byteArrayClass\r\n
e is ShortArray -> PrimitiveClasses.shortArrayClass\r\n e is IntArray ->
PrimitiveClasses.intArrayClass\r\n e is LongArray -> PrimitiveClasses.longArrayClass\r\n e is
FloatArray -> PrimitiveClasses.floatArrayClass\r\n e is DoubleArray ->
PrimitiveClasses.doubleArrayClass\r\n e is KClass<*> -> KClass::class\r\n e is Array<*> ->
PrimitiveClasses.arrayClass\r\n else -> {\r\n val constructor =
js("Object").getPrototypeOf(e).constructor\r\n when {\r\n constructor === js("Object")
-> PrimitiveClasses.anyClass\r\n constructor === js("Error") -> PrimitiveClasses.throwableClass\r\n
else -> {\r\n val jsClass: JsClass<T> = constructor\r\n
getOrCreateKClass(jsClass)\r\n }\r\n }\r\n }\r\n }\r\n }\r\n
}.unsafeCast<KClass<T>>()\r\n\r\nprivate fun <T : Any> getOrCreateKClass(jClass: JsClass<T>): KClass<T> {\r\n
if (jClass === js("String")) return PrimitiveClasses.stringClass.unsafeCast<KClass<T>>()\r\n\r\n val metadata
= jClass.asDynamic().`$metadata$\r\n\r\n return if (metadata != null) {\r\n if (metadata.`$kClass$` == null)
{\r\n val kClass = SimpleKClassImpl(jClass)\r\n metadata.`$kClass$` = kClass\r\n kClass\r\n
} else {\r\n metadata.`$kClass$`\r\n }\r\n } else {\r\n SimpleKClassImpl(jClass)\r\n
}\r\n }\r\n\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n *
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*/\r\n\r\npackage kotlin.js\r\n\r\n/**\r\n * Exposes the JavaScript [RegExp
object](https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Global_Objects/RegExp) to Kotlin.\r\n
*/\r\n@Suppress("NOT_DOCUMENTED")\r\npublic external class RegExp(pattern: String, flags: String? =
definedExternally) {\r\n\r\n public fun test(str: String): Boolean\r\n\r\n public fun exec(str: String):
RegExpMatch?\r\n\r\n public override fun toString(): String\r\n\r\n /**\r\n * The lastIndex is a read/write
integer property of regular expressions that specifies the index at which to start the next match.\r\n */\r\n public
var lastIndex: Int\r\n\r\n public val global: Boolean\r\n public val ignoreCase: Boolean\r\n public val multiline:
Boolean\r\n}\r\n\r\n/**\r\n * Resets the regular expression so that subsequent [RegExp.test] and [RegExp.exec] calls
will match starting with the beginning of the input string.\r\n */\r\n public fun RegExp.reset() {\r\n lastIndex =
0\r\n}\r\n\r\n// TODO: Inherit from array or introduce asArray() extension\r\n/**\r\n * Represents the return value of
[RegExp.exec].\r\n */\r\n@Suppress("NOT_DOCUMENTED")\r\npublic external interface RegExpMatch {\r\n
public val index: Int\r\n public val input: String\r\n public val length: Int\r\n}\r\n\r\n/**\r\n * Returns the entire
text matched by [RegExp.exec] if the [index] parameter is 0, or the text matched by the capturing parenthesis\r\n * at
the given index.\r\n */\r\n public inline operator fun RegExpMatch.get(index: Int): String? =
asDynamic()[index]\r\n\r\n/**\r\n * Converts the result of [RegExp.exec] to an array where the first element

```



NumberFormatException if the string is not a valid representation of a number. `\r\n * @throws`

IllegalArgumentException when [radix] is not a valid radix for string to number conversion. `\r\n */\r\npublic actual`

fun String.toLong(radix: Int): Long = toLongOrNull(radix) ?: numberFormatError(this)\r\n\r\n/\*\*\r\n \* Parses the string as a [Double] number and returns the result.\r\n \* @throws NumberFormatException if the string is not a valid representation of a number.\r\n \*/\r\npublic actual fun String.toDouble(): Double =

(+(this.asDynamic()))\r\n .unsafeCast<Double>().also { \r\n if (it.isNaN() && !this.isNaN() || it == 0.0 && this.isBlank())\r\n numberFormatError(this)\r\n }\r\n\r\n/\*\*\r\n \* Parses the string as a [Float] number and returns the result.\r\n \* @throws NumberFormatException if the string is not a valid representation of a number.\r\n \*/\r\n@kotlin.internal.InlineOnly\r\npublic actual inline fun String.toFloat(): Float =

toDouble().unsafeCast<Float>()\r\n\r\n/\*\*\r\n \* Parses the string as a [Double] number and returns the result\r\n \* or `null` if the string is not a valid representation of a number.\r\n \*/\r\npublic actual fun String.toDoubleOrNull(): Double? =

(+(this.asDynamic()))\r\n .unsafeCast<Double>().takeIf { \r\n !it.isNaN() && !this.isNaN() || it == 0.0 && this.isBlank()\r\n }\r\n\r\n/\*\*\r\n \* Parses the string as a [Float] number and returns the result\r\n \* or `null` if the string is not a valid representation of a number.\r\n \*/\r\n@kotlin.internal.InlineOnly\r\npublic actual inline fun String.toFloatOrNull(): Float? =

toDoubleOrNull().unsafeCast<Float?>()\r\n\r\n/\*\*\r\n \* Returns a string representation of this [Byte] value in the specified [radix].\r\n \* @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\r\n \*/\r\n@SinceKotlin("1.2")\r\n@kotlin.internal.InlineOnly\r\npublic actual inline fun Byte.toString(radix: Int): String =

this.toInt().toString(radix)\r\n\r\n/\*\*\r\n \* Returns a string representation of this [Short] value in the specified [radix].\r\n \* @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\r\n \*/\r\n@SinceKotlin("1.2")\r\n@kotlin.internal.InlineOnly\r\npublic actual inline fun Short.toString(radix: Int): String =

this.toInt().toString(radix)\r\n\r\n/\*\*\r\n \* Returns a string representation of this [Int] value in the specified [radix].\r\n \* @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.\r\n \*/\r\n@SinceKotlin("1.2")\r\npublic actual fun Int.toString(radix: Int): String =

asDynamic().toString(checkRadix(radix))\r\n\r\nprivate fun String.isNaN(): Boolean = when (this.toLowerCase()) { \r\n "nan", "+nan", "-nan" -> true\r\n else -> false\r\n}\r\n\r\n/\*\*\r\n \* Checks whether the given [radix] is valid radix for string to number and number to string conversion.\r\n \*/\r\n@PublishedApi\r\ninternal actual fun checkRadix(radix: Int): Int { \r\n if (radix !in 2..36) { \r\n throw IllegalArgumentException("\r\nradix \$radix was not in valid range 2..36")\r\n } \r\n return radix\r\n}\r\n\r\ninternal actual fun digitOf(char: Char, radix: Int): Int = when { \r\n char >= '0' && char <= '9' -> char - '0'\r\n char >= 'A' && char <= 'Z' -> char - 'A' + 10\r\n char >= 'a' && char <= 'z' -> char - 'a' + 10\r\n else -> -1\r\n}.let { if (it >= radix) -1 else it }\r\n\r\n",\*/\r\n \* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n \* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n \*/\r\n\r\npackage kotlin.text\r\n\r\nimport kotlin.js.RegExp\r\n\r\n/\*\*\r\n \* Provides enumeration values to use to set regular expression options.\r\n \*/\r\npublic actual enum class RegexOptions(val value: String) { \r\n /\*\* Enables case-insensitive matching. \*/\r\n IGNORE\_CASE("i"), \r\n /\*\* Enables multiline mode. \*/\r\n MULTILINE("m")\r\n}\r\n\r\n/\*\* Represents the results from a single capturing group within a [MatchResult] of [Regex].\r\n \* @param value The value of captured group.\r\n \*/\r\npublic actual data class MatchGroup(actual val value: String)\r\n\r\n/\*\* Represents a compiled regular expression.\r\n \* Provides functions to match strings in text with a pattern, replace the found occurrences and split text around matches.\r\n \* For pattern syntax reference see [MDN RegExp](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\_Objects/RegExp#Special\_characters\_meaning\_in\_regular\_expressions)\r\n \*/\r\n \* and [http://www.w3schools.com/jsref/jsref\_obj\_regexp.asp](https://www.w3schools.com/jsref/jsref\_obj\_regexp.asp).\r\n \*/\r\n \* @constructor Creates a regular expression from the specified [pattern] string and the specified set of [options].\r\n \*/\r\npublic actual class Regex actual constructor(pattern: String, options: Set<RegexOption>) { \r\n /\*\* Creates a regular expression from the specified [pattern] string and the specified single [option].

```

*/\r\n public actual constructor(pattern: String, option: RegexOptions) : this(pattern, setOf(option))\r\n\r\n /**
Creates a regular expression from the specified [pattern] string and the default options. */\r\n public actual
constructor(pattern: String) : this(pattern, emptySet())\r\n\r\n\r\n /** The pattern string of this regular expression.
*/\r\n public actual val pattern: String = pattern\r\n\r\n /** The set of options that were used to create this regular
expression. */\r\n public actual val options: Set<RegexOption> = options.toSet()\r\n private val nativePattern:
RegExp = RegExp(pattern, options.map { it.value }.joinToString(separator = "\\") + "g")\r\n\r\n /** Indicates
whether the regular expression matches the entire [input]. */\r\n public actual infix fun matches(input:
CharSequence): Boolean {\r\n nativePattern.reset()\r\n val match = nativePattern.exec(input.toString())\r\n
return match != null && match.index == 0 && nativePattern.lastIndex == input.length\r\n }\r\n\r\n /**
Indicates whether the regular expression can find at least one match in the specified [input]. */\r\n public actual
fun containsMatchIn(input: CharSequence): Boolean {\r\n nativePattern.reset()\r\n return
nativePattern.test(input.toString())\r\n }\r\n\r\n\r\n /** Returns the first match of a regular expression in the [input],
beginning at the specified [startIndex].\r\n * \r\n * @param startIndex An index to start search with, by default
0. Must be not less than zero and not greater than `input.length()`\r\n * @return An instance of [MatchResult] if
match was found or `null` otherwise.\r\n * \r\n
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\n public actual fun find(input:
CharSequence, startIndex: Int = 0): MatchResult? = nativePattern.findNext(input.toString(), startIndex)\r\n\r\n
/**\r\n * Returns a sequence of all occurrences of a regular expression within the [input] string, beginning at the
specified [startIndex].\r\n * \r\n * @sample samples.text.Regexp.findAll\r\n * \r\n
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\n public actual fun findAll(input:
CharSequence, startIndex: Int = 0): Sequence<MatchResult> =\r\n generateSequence({ find(input, startIndex) },
{ match -> match.next() })\r\n\r\n /**\r\n * Attempts to match the entire [input] CharSequence against the
pattern.\r\n * \r\n * @return An instance of [MatchResult] if the entire input matches or `null` otherwise.\r\n
*/\r\n public actual fun matchEntire(input: CharSequence): MatchResult? {\r\n if (pattern.startsWith('^') &&
pattern.endsWith('$'))\r\n return find(input)\r\n else\r\n return
Regex("^${pattern.trimStart('^').trimEnd('$')}\$", options).find(input)\r\n }\r\n\r\n\r\n /**\r\n * Replaces all
occurrences of this regular expression in the specified [input] string with specified [replacement] expression.\r\n
\r\n * \r\n * @param replacement A replacement expression that can include substitutions. See
[String.prototype.replace](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String/replace) function docs for details.\r\n
*/\r\n public actual fun replace(input: CharSequence, replacement: String): String = input.toString().nativeReplace(nativePattern,
replacement)\r\n\r\n\r\n /**\r\n * Replaces all occurrences of this regular expression in the specified [input] string
with the result of\r\n * the given function [transform] that takes [MatchResult] and returns a string to be used as
a\r\n * replacement for that match.\r\n * \r\n public actual inline fun replace(input: CharSequence, transform:
(MatchResult) -> CharSequence): String {\r\n var match = find(input)\r\n if (match == null) return
input.toString()\r\n\r\n var lastStart = 0\r\n val length = input.length\r\n val sb =
StringBuilder(length)\r\n do {\r\n val foundMatch = match!!\r\n sb.append(input, lastStart,
foundMatch.range.start)\r\n sb.append(transform(foundMatch))\r\n lastStart =
foundMatch.range.endInclusive + 1\r\n match = foundMatch.next()\r\n } while (lastStart < length &&
match != null)\r\n\r\n if (lastStart < length) {\r\n sb.append(input, lastStart, length)\r\n }\r\n\r\n
return sb.toString()\r\n }\r\n\r\n\r\n /**\r\n * Replaces the first occurrence of this regular expression in the
specified [input] string with specified [replacement] expression.\r\n * \r\n * @param replacement A replacement
expression that can include substitutions. See [String.prototype.replace](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String/replace) function docs for details.\r\n
*/\r\n public actual fun replaceFirst(input: CharSequence, replacement: String): String {\r\n val nonGlobalOptions =
options.map { it.value }.joinToString(separator = "\\")\r\n return input.toString().nativeReplace(RegExp(pattern,
nonGlobalOptions), replacement)\r\n }\r\n\r\n\r\n /**\r\n * Splits the [input] CharSequence around matches of
this regular expression.\r\n * \r\n * @param limit Non-negative value specifying the maximum number of

```

```

substrings the string can be split to.\r\n * Zero by default means no limit is set.\r\n */\r\n
@Suppress(\\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\\")\r\n public actual fun split(input:
CharSequence, limit: Int = 0): List<String> {\r\n require(limit >= 0) { \\"Limit must be non-negative, but was
$limit\\" }\r\n val matches = findAll(input).let { if (limit == 0) it else it.take(limit - 1) }\r\n val result =
mutableListOf<String>()\r\n var lastStart = 0\r\n for (match in matches) {\r\n
result.add(input.subSequence(lastStart, match.range.start).toString())\r\n lastStart = match.range.endInclusive
+ 1\r\n }\r\n result.add(input.subSequence(lastStart, input.length).toString())\r\n return result\r\n
}\r\n\r\n\r\n /**\r\n * Returns the string representation of this regular expression, namely the [pattern] of this
regular expression.\r\n */\r\n * Note that another regular expression constructed from the same pattern string
may have different [options]\r\n * and may match strings differently.\r\n */\r\n public override fun toString():
String = nativePattern.toString()\r\n\r\n actual companion object {\r\n /**\r\n * Returns a regular
expression that matches the specified [literal] string literally.\r\n * No characters of that string will have special
meaning when searching for an occurrence of the regular expression.\r\n */\r\n public actual fun
fromLiteral(literal: String): Regex = Regex(escape(literal))\r\n /**\r\n * Returns a regular expression
pattern string that matches the specified [literal] string literally.\r\n * No characters of that string will have
special meaning when searching for an occurrence of the regular expression.\r\n */\r\n public actual fun
escape(literal: String): String = literal.nativeReplace(patternEscape, '\\\\$&\\')\r\n /**\r\n * Returns a
literal replacement expression for the specified [literal] string.\r\n * No characters of that string will have
special meaning when it is used as a replacement string in [Regex.replace] function.\r\n */\r\n public actual
fun escapeReplacement(literal: String): String = literal.nativeReplace(replacementEscape, '\\$$$')\r\n\r\n
private val patternEscape = Regex("\\\\\\"[-\\\\\\^$*+?.()|\\\\\\{ }]"\\\\\\\\", "\\\\"g\\")\r\n private val replacementEscape =
Regex("\\\\\\"\\\\\\$"\\\\\\\\", "\\\\"g\\")\r\n }\r\n}\r\n\r\n\r\n/** Creates a regular expression from the specified [pattern] string
and the specified single [option]. */\r\n@Deprecated(\\"Provided for binary compatibility\\", level =
DeprecationLevel.HIDDEN)\r\n@JsName(\\"Regex_sb3q2\\\$\\")\r\npublic fun Regex_0(pattern: String, option:
RegexOption): Regex = Regex(pattern, setOf(option))\r\n\r\n\r\n/** Creates a regular expression from the specified
[pattern] string and the default options. */\r\n@Deprecated(\\"Provided for binary compatibility\\", level =
DeprecationLevel.HIDDEN)\r\n@JsName(\\"Regex_61zpoel\\\$\\")\r\npublic fun Regex_1(pattern: String): Regex =
Regex(pattern, emptySet())\r\n\r\n\r\n\r\n\r\nprivate fun RegExp.findNext(input: String, from: Int): MatchResult?
{\r\n this.lastIndex = from\r\n val match = exec(input)\r\n if (match == null) return null\r\n val range =
match.index..lastIndex - 1\r\n return object : MatchResult {\r\n override val range: IntRange = range\r\n
 override val value: String\r\n get() = match[0]!\r\n override val groups: MatchGroupCollection =
object : MatchGroupCollection, AbstractCollection<MatchGroup?>() {\r\n override val size: Int get() =
match.length\r\n override fun iterator(): Iterator<MatchGroup?> = indices.asSequence().map { this[it]
}.iterator()\r\n override fun get(index: Int): MatchGroup? = match[index]?.let { MatchGroup(it) }\r\n
}\r\n\r\n\r\n private var groupValues_: List<String>? = null\r\n override val groupValues:
List<String>\r\n get() {\r\n if (groupValues_ == null) {\r\n groupValues_ = object :
AbstractList<String>() {\r\n override val size: Int get() = match.length\r\n override fun
get(index: Int): String = match[index] ?: \\\\"\\r\n }\r\n }\r\n return groupValues_!\r\n
}\r\n\r\n override fun next(): MatchResult? = this@findNext.findNext(input, if (range.isEmpty())
range.start + 1 else range.endInclusive + 1)\r\n }\r\n}\r\n}\r\n\r\n", "*/\r\n * Copyright 2010-2018 JetBrains s.r.o. and
Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that
can be found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.text\r\n\r\nimport
kotlin.js.RegExp\r\n\r\n@kotlin.internal.InlineOnly\r\ninternal actual inline fun String.nativeIndexOf(ch: Char,
fromIndex: Int): Int = nativeIndexOf(ch.toString(), fromIndex)\r\n\r\n@kotlin.internal.InlineOnly\r\ninternal actual
inline fun String.nativeLastIndexOf(ch: Char, fromIndex: Int): Int = nativeLastIndexOf(ch.toString(),
fromIndex)\r\n\r\n\r\n\r\n/** Returns `true` if this string starts with the specified prefix.\r\n */\r\n
*/\r\n@Suppress(\\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\\")\r\npublic actual fun
String.startsWith(prefix: String, ignoreCase: Boolean = false): Boolean {\r\n if (!ignoreCase)\r\n return

```

```

nativeStartsWith(prefix, 0)\r\n else\r\n return regionMatches(0, prefix, 0, prefix.length,
ignoreCase)\r\n}\r\n\r\n/**\r\n * Returns `true` if a substring of this string starting at the specified offset [startIndex]
starts with the specified prefix.\r\n
*/\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\npublic actual fun
String.startsWith(prefix: String, startIndex: Int, ignoreCase: Boolean = false): Boolean {\r\n if (!ignoreCase)\r\n return nativeStartsWith(prefix, startIndex)\r\n else\r\n return regionMatches(startIndex, prefix, 0,
prefix.length, ignoreCase)\r\n}\r\n\r\n/**\r\n * Returns `true` if this string ends with the specified suffix.\r\n
*/\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\npublic actual fun
String.endsWith(suffix: String, ignoreCase: Boolean = false): Boolean {\r\n if (!ignoreCase)\r\n return
nativeEndsWith(suffix)\r\n else\r\n return regionMatches(length - suffix.length, suffix, 0, suffix.length,
ignoreCase)\r\n}\r\n\r\n\r\npublic fun String.matches(regex: String): Boolean {\r\n val result =
this.match(regex)\r\n return result != null && result.size != 0\r\n}\r\n\r\n\r\npublic actual fun
CharSequence.isBlank(): Boolean = length == 0 || (if (this is String) this else
this.toString()).matches("^[\\s\\xA0]+$")\r\n\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARG
UMENTS")\r\npublic actual fun String?.equals(other: String?, ignoreCase: Boolean = false): Boolean =\r\n if
(this == null)\r\n other == null\r\n else if (!ignoreCase)\r\n this == other\r\n else\r\n other != null
&& this.toLowerCase() ==
other.toLowerCase()\r\n\r\n\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\npublic
actual fun CharSequence.regionMatches(thisOffset: Int, other: CharSequence, otherOffset: Int, length: Int,
ignoreCase: Boolean = false): Boolean =\r\n regionMatchesImpl(thisOffset, other, otherOffset, length,
ignoreCase)\r\n\r\n\r\n/**\r\n * Returns a copy of this string having its first letter uppercased, or the original
string,\r\n * if it's empty or already starts with an upper case letter.\r\n
*/\r\n * @sample
samples.text.Strings.capitalize\r\n
*/\r\npublic actual fun String.capitalize(): String {\r\n return if (isEmpty())
substring(0, 1).toUpperCase() + substring(1) else this\r\n}\r\n\r\n\r\n/**\r\n * Returns a copy of this string having its
first letter lowercased, or the original string,\r\n * if it's empty or already starts with a lower case letter.\r\n
*/\r\n * @sample
samples.text.Strings.decapitalize\r\n
*/\r\npublic actual fun String.decapitalize(): String {\r\n return if
(isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this\r\n}\r\n\r\n\r\n/**\r\n * Returns a string
containing this char sequence repeated [n] times.\r\n
*/\r\n * @throws [IllegalArgumentException] when n < 0.\r\n
*/\r\npublic actual fun CharSequence.repeat(n: Int): String {\r\n require(n >= 0) { "Count 'n' must be non-
negative, but was $n." }\r\n return when (n) {\r\n 0 -> ""\r\n 1 -> this.toString()\r\n else -> {\r\n
 var result = ""\r\n if (!isEmpty()) {\r\n var s = this.toString()\r\n var count = n\r\n
 while (true) {\r\n if ((count and 1) == 1) {\r\n result += s\r\n }\r\n
 count = count ushr 1\r\n if (count == 0) {\r\n break\r\n }\r\n
 s +=
s\r\n }\r\n return result\r\n }\r\n
}\r\n}\r\n\r\n\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\npublic actual fun
String.replace(oldValue: String, newValue: String, ignoreCase: Boolean = false): String =\r\n
nativeReplace(Regex.escape(Regex.escape(oldValue)), if (ignoreCase) "gi" else "g"),
Regex.escapeReplacement(newValue))\r\n\r\n\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUME
NTS")\r\npublic actual fun String.replace(oldChar: Char, newChar: Char, ignoreCase: Boolean = false): String =\r\n
nativeReplace(Regex.escape(Regex.escape(oldChar.toString())), if (ignoreCase) "gi" else "g"),
newChar.toString())\r\n\r\n\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\r\npublic
actual fun String.replaceFirst(oldValue: String, newValue: String, ignoreCase: Boolean = false): String =\r\n
nativeReplace(Regex.escape(Regex.escape(oldValue)), if (ignoreCase) "i" else ""),
Regex.escapeReplacement(newValue))\r\n\r\n\r\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUME
NTS")\r\npublic actual fun String.replaceFirst(oldChar: Char, newChar: Char, ignoreCase: Boolean = false): String
=\r\n nativeReplace(Regex.escape(Regex.escape(oldChar.toString())), if (ignoreCase) "i" else ""),
newChar.toString())\r\n
*/\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the

```

```

license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.text\r\n\r\n\r\npublic actual interface Appendable {\r\n
public actual fun append(csq: CharSequence?): Appendable\r\n public actual fun append(csq: CharSequence?,
start: Int, end: Int): Appendable\r\n public actual fun append(c: Char): Appendable\r\n}\r\n\r\n\r\npublic actual class
StringBuilder(content: String = "") : Appendable, CharSequence {\r\n actual constructor(capacity: Int) : this()
{\r\n\r\n actual constructor(content: CharSequence) : this(content.toString()) {\r\n\r\n actual constructor() :
this("")\r\n\r\n private var string: String = content\r\n\r\n actual override val length: Int\r\n get() =
string.asDynamic().length\r\n\r\n actual override fun get(index: Int): Char =\r\n string.getOrElse(index) {
throw IndexOutOfBoundsException("index: $index, length: $length")}\r\n\r\n actual override fun
subSequence(startIndex: Int, endIndex: Int): CharSequence = string.substring(startIndex, endIndex)\r\n\r\n actual
override fun append(c: Char): StringBuilder {\r\n string += c\r\n return this\r\n }\r\n\r\n actual override
fun append(csq: CharSequence?): StringBuilder {\r\n string += csq.toString()\r\n return this\r\n }\r\n\r\n
actual override fun append(csq: CharSequence?, start: Int, end: Int): StringBuilder {\r\n string +=
csq.toString().substring(start, end)\r\n return this\r\n }\r\n\r\n actual fun append(obj: Any?): StringBuilder
{\r\n string += obj.toString()\r\n return this\r\n }\r\n\r\n actual fun reverse(): StringBuilder {\r\n
string = string.asDynamic().split("").reverse().join("")\r\n return this\r\n }\r\n\r\n /**\r\n * Clears the
content of this string builder making it empty.\r\n */\r\n * @sample samples.text.Strings.clearStringBuilder\r\n
*/\r\n * @SinceKotlin("1.3")\r\n public fun clear(): StringBuilder {\r\n string = ""\r\n return this\r\n
}\r\n\r\n override fun toString(): String = string\r\n}\r\n\r\n\r\n\r\n/**\r\n * Clears the content of this string builder
making it empty.\r\n */\r\n * @sample samples.text.Strings.clearStringBuilder\r\n
*/\r\n @SinceKotlin("1.3")\r\n @Suppress("EXTENSION_SHADOWED_BY_MEMBER",
"NOTHING_TO_INLINE")\r\n public actual inline fun StringBuilder.clear(): StringBuilder =
this.clear()\r\n\r\n\r\n/**\r\n * The exception thrown when a character encoding or decoding error occurs.\r\n
*/\r\n @SinceKotlin("1.3")\r\n @ExperimentalStdlibApi\r\n public actual open class
CharacterCodingException(message: String?) : Exception(message) {\r\n actual constructor() :
this(null)\r\n}\r\n\r\n", "*/\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\n\r\npackage kotlin.text\r\n\r\n\r\n/** Returns the negative [size] if
[throwOnMalformed] is false, throws [CharacterCodingException] otherwise.
*/\r\n @UseExperimental(ExperimentalStdlibApi::class)\r\n private fun malformed(size: Int, index: Int,
throwOnMalformed: Boolean): Int {\r\n if (throwOnMalformed) throw CharacterCodingException("Malformed
sequence starting at ${index - 1}")\r\n return -size\r\n}\r\n\r\n\r\n/** Returns code point corresponding to UTF-
16 surrogate pair,\r\n * where the first of the pair is the [high] and the second is in the [string] at the [index].\r\n *
Returns zero if the pair is malformed and [throwOnMalformed] is false.\r\n */\r\n * @throws
CharacterCodingException if the pair is malformed and [throwOnMalformed] is true.\r\n */\r\n private fun
codePointFromSurrogate(string: String, high: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {\r\n
if (high !in 0xD800..0xDBFF || index >= endIndex) {\r\n return malformed(0, index, throwOnMalformed)\r\n
}\r\n val low = string[index].toInt()\r\n if (low !in 0xDC00..0xDFFF) {\r\n return malformed(0, index,
throwOnMalformed)\r\n }\r\n return 0x10000 + ((high and 0x3FF) shl 10) or (low and 0x3FF)\r\n}\r\n\r\n\r\n\r\n
/** Returns code point corresponding to UTF-8 sequence of two bytes,\r\n * where the first byte of the sequence is the
[byte1] and the second byte is in the [bytes] array at the [index].\r\n * Returns zero if the sequence is malformed and
[throwOnMalformed] is false.\r\n */\r\n * @throws CharacterCodingException if the sequence of two bytes is
malformed and [throwOnMalformed] is true.\r\n */\r\n private fun codePointFrom2(bytes: ByteArray, byte1: Int,
index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {\r\n if (byte1 and 0x1E == 0 || index >= endIndex)
{\r\n return malformed(0, index, throwOnMalformed)\r\n }\r\n val byte2 = bytes[index].toInt()\r\n if
(byte2 and 0xC0 != 0x80) {\r\n return malformed(0, index, throwOnMalformed)\r\n }\r\n return (byte1 shl
6) xor byte2 xor 0xF80\r\n}\r\n\r\n\r\n\r\n/** Returns code point corresponding to UTF-8 sequence of three bytes,\r\n
* where the first byte of the sequence is the [byte1] and the others are in the [bytes] array starting from the
[index].\r\n * Returns a non-positive value indicating number of bytes from [bytes] included in malformed

```



```

sequence\r\n * if the sequence is malformed and [throwOnMalformed] is false.\r\n * @throws
CharacterCodingException if the sequence of three bytes is malformed and [throwOnMalformed] is true.\r\n
*/\r\nprivate fun codePointFrom3(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed:
Boolean): Int {\r\n if (index >= endIndex) {\r\n return malformed(0, index, throwOnMalformed)\r\n }\r\n val byte2 = bytes[index].toInt()\r\n if (byte1 and 0xF == 0) {\r\n if (byte2 and 0xE0 != 0xA0) {\r\n //
Non-shortest form\r\n return malformed(0, index, throwOnMalformed)\r\n } else if (byte1 and
0xF == 0xD) {\r\n if (byte2 and 0xE0 != 0x80) {\r\n // Surrogate code point\r\n return
malformed(0, index, throwOnMalformed)\r\n } else if (byte2 and 0xC0 != 0x80) {\r\n return
malformed(0, index, throwOnMalformed)\r\n }\r\n\r\n if (index + 1 == endIndex) {\r\n return malformed(1,
index, throwOnMalformed)\r\n }\r\n val byte3 = bytes[index + 1].toInt()\r\n if (byte3 and 0xC0 != 0x80) {\r\n
 return malformed(1, index, throwOnMalformed)\r\n }\r\n\r\n return (byte1 shl 12) xor (byte2 shl 6) xor byte3
xor -0x1E080\r\n}\r\n\r\n/**\r\n * Returns code point corresponding to UTF-8 sequence of four bytes,\r\n * where
the first byte of the sequence is the [byte1] and the others are in the [bytes] array starting from the [index].\r\n *
Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence\r\n * if the
sequence is malformed and [throwOnMalformed] is false.\r\n * @throws CharacterCodingException if the
sequence of four bytes is malformed and [throwOnMalformed] is true.\r\n */\r\nprivate fun codePointFrom4(bytes:
ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {\r\n if (index >= endIndex)
{\r\n malformed(0, index, throwOnMalformed)\r\n }\r\n val byte2 = bytes[index].toInt()\r\n if (byte1
and 0xF == 0x0) {\r\n if (byte2 and 0xF0 <= 0x80) {\r\n // Non-shortest form\r\n return
malformed(0, index, throwOnMalformed)\r\n } else if (byte1 and 0xF == 0x4) {\r\n if (byte2 and
0xF0 != 0x80) {\r\n // Out of Unicode code points domain (larger than U+10FFFF)\r\n return
malformed(0, index, throwOnMalformed)\r\n } else if (byte1 and 0xF > 0x4) {\r\n return
malformed(0, index, throwOnMalformed)\r\n } else if (byte2 and 0xC0 != 0x80) {\r\n return malformed(0,
index, throwOnMalformed)\r\n }\r\n\r\n if (index + 1 == endIndex) {\r\n return malformed(1, index,
throwOnMalformed)\r\n }\r\n val byte3 = bytes[index + 1].toInt()\r\n if (byte3 and 0xC0 != 0x80) {\r\n
 return malformed(1, index, throwOnMalformed)\r\n }\r\n\r\n if (index + 2 == endIndex) {\r\n return
malformed(2, index, throwOnMalformed)\r\n }\r\n val byte4 = bytes[index + 2].toInt()\r\n if (byte4 and 0xC0
!= 0x80) {\r\n return malformed(2, index, throwOnMalformed)\r\n }\r\n\r\n return (byte1 shl 18) xor (byte2 shl
12) xor (byte3 shl 6) xor byte4 xor 0x381F80\r\n}\r\n\r\n/**\r\n * Maximum number of bytes needed to encode a
single char.\r\n * Code points in `0..0x7F` are encoded in a single byte.\r\n * Code points in `0x80..0x7FF` are
encoded in two bytes.\r\n * Code points in `0x800..0xD7FF` or in `0xE000..0xFFFF` are encoded in three bytes.\r\n
* Surrogate code points in `0xD800..0xDFFF` are not Unicode scalar values, therefore aren't encoded.\r\n * Code
points in `0x10000..0x10FFFF` are represented by a pair of surrogate `Char`s and are encoded in four bytes.\r\n
*/\r\nprivate const val MAX_BYTES_PER_CHAR = 3\r\n\r\n/**\r\n * The byte sequence a malformed UTF-16
char sequence is replaced by.\r\n */\r\nprivate val REPLACEMENT_BYTE_SEQUENCE: ByteArray =
byteArrayOf(0xEF.toByte(), 0xBF.toByte(), 0xBD.toByte())\r\n\r\n/**\r\n * Encodes the [string] using UTF-8 and
returns the resulting [ByteArray].\r\n * @param string the string to encode.\r\n * @param startIndex the start
offset (inclusive) of the substring to encode.\r\n * @param endIndex the end offset (exclusive) of the substring to
encode.\r\n * @param throwOnMalformed whether to throw on malformed char sequence or replace by the
[REPLACEMENT_BYTE_SEQUENCE].\r\n * @throws CharacterCodingException if the char sequence is
malformed and [throwOnMalformed] is true.\r\n */\r\ninternal fun encodeUtf8(string: String, startIndex: Int,
endIndex: Int, throwOnMalformed: Boolean): ByteArray {\r\n require(startIndex >= 0 && endIndex <=
string.length && startIndex <= endIndex)\r\n val bytes = ByteArray((endIndex - startIndex) *
MAX_BYTES_PER_CHAR)\r\n var byteIndex = 0\r\n var charIndex = startIndex\r\n while (charIndex <
endIndex) {\r\n val code = string[charIndex++].toInt()\r\n when {\r\n code < 0x80 ->\r\n
bytes[byteIndex++] = code.toByte()\r\n code < 0x800 -> {\r\n bytes[byteIndex++] = ((code shr 6) or
0xC0).toByte()\r\n bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()\r\n }\r\n code <
0xD800 || code >= 0xE000 -> {\r\n bytes[byteIndex++] = ((code shr 12) or 0xE0).toByte()\r\n

```

```

bytes[byteIndex++] = (((code shr 6) and 0x3F) or 0x80).toByte()\r\n bytes[byteIndex++] = ((code and
0x3F) or 0x80).toByte()\r\n }\r\n else -> { // Surrogate char value\r\n val codePoint =
codePointFromSurrogate(string, code, charIndex, endIndex, throwOnMalformed)\r\n if (codePoint <= 0)
{\r\n bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[0]\r\n
bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[1]\r\n bytes[byteIndex++] =
REPLACEMENT_BYTE_SEQUENCE[2]\r\n } else {\r\n bytes[byteIndex++] = ((codePoint
shr 18) or 0xF0).toByte()\r\n bytes[byteIndex++] = (((codePoint shr 12) and 0x3F) or 0x80).toByte()\r\n
bytes[byteIndex++] = (((codePoint shr 6) and 0x3F) or 0x80).toByte()\r\n bytes[byteIndex++]
= ((codePoint and 0x3F) or 0x80).toByte()\r\n charIndex++\r\n }\r\n }\r\n }\r\n
}\r\n\r\n return if (bytes.size == byteIndex) bytes else bytes.copyOf(byteIndex)\r\n}\r\n\r\n**\r\n * The character
a malformed UTF-8 byte sequence is replaced by.\r\n */\r\nprivate const val REPLACEMENT_CHAR =
"\uFFFFD\r\n\r\n**\r\n * Decodes the UTF-8 [bytes] array and returns the resulting [String].\r\n * @param
bytes the byte array to decode.\r\n * @param startIndex the start offset (inclusive) of the array to be decoded.\r\n *
@param endIndex the end offset (exclusive) of the array to be encoded.\r\n * @param throwOnMalformed whether
to throw on malformed byte sequence or replace by the [REPLACEMENT_CHAR].\r\n * @throws
CharacterCodingException if the array is malformed UTF-8 byte sequence and [throwOnMalformed] is true.\r\n
*/\r\ninternal fun decodeUtf8(bytes: ByteArray, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): String
{\r\n require(startIndex >= 0 && endIndex <= bytes.size && startIndex <= endIndex)\r\n\r\n val byteIndex =
startIndex\r\n val stringBuilder = StringBuilder()\r\n\r\n while (byteIndex < endIndex) {\r\n val byte =
bytes[byteIndex++].toInt()\r\n when {\r\n byte >= 0 ->\r\n
stringBuilder.append(byte.toChar())\r\n byte shr 5 == -2 -> {\r\n val code = codePointFrom2(bytes,
byte, byteIndex, endIndex, throwOnMalformed)\r\n if (code <= 0) {\r\n
stringBuilder.append(REPLACEMENT_CHAR)\r\n byteIndex += -code\r\n } else {\r\n
stringBuilder.append(code.toChar())\r\n byteIndex += 1\r\n }\r\n }\r\n byte
shr 4 == -2 -> {\r\n val code = codePointFrom3(bytes, byte, byteIndex, endIndex, throwOnMalformed)\r\n
if (code <= 0) {\r\n stringBuilder.append(REPLACEMENT_CHAR)\r\n byteIndex
+= -code\r\n } else {\r\n stringBuilder.append(code.toChar())\r\n byteIndex +=
2\r\n }\r\n }\r\n byte shr 3 == -2 -> {\r\n val code = codePointFrom4(bytes, byte,
byteIndex, endIndex, throwOnMalformed)\r\n if (code <= 0) {\r\n
stringBuilder.append(REPLACEMENT_CHAR)\r\n byteIndex += -code\r\n } else {\r\n
val high = (code - 0x10000) shr 10 or 0xD800\r\n val low = (code and 0x3FF) or 0xDC00\r\n
stringBuilder.append(high.toChar())\r\n stringBuilder.append(low.toChar())\r\n
byteIndex += 3\r\n }\r\n }\r\n } else -> {\r\n malformed(0, byteIndex,
throwOnMalformed)\r\n stringBuilder.append(REPLACEMENT_CHAR)\r\n }\r\n }\r\n
}\r\n\r\n return stringBuilder.toString()\r\n}"/**\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage
kotlin.time\r\n\r\n@SinceKotlin("1.3")\r\n@ExperimentalTime\r\npublic actual enum class DurationUnit(internal
val scale: Double) {\r\n /**\r\n * Time unit representing one nanosecond, which is 1/1000 of a microsecond.\r\n
*/\r\n NANOSECONDS(1e0),\r\n /**\r\n * Time unit representing one microsecond, which is 1/1000 of a
millisecond.\r\n */\r\n MICROSECONDS(1e3),\r\n /**\r\n * Time unit representing one millisecond, which
is 1/1000 of a second.\r\n */\r\n MILLISECONDS(1e6),\r\n /**\r\n * Time unit representing one second.\r\n
*/\r\n SECONDS(1e9),\r\n /**\r\n * Time unit representing one minute.\r\n */\r\n MINUTES(60e9),\r\n
/**\r\n * Time unit representing one hour.\r\n */\r\n HOURS(3600e9),\r\n /**\r\n * Time unit
representing one day, which is always equal to 24 hours.\r\n */\r\n
DAYS(86400e9);\r\n}\r\n\r\n@SinceKotlin("1.3")\r\n@ExperimentalTime\r\ninternal actual fun
convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double {\r\n val
sourceCompareTarget = sourceUnit.scale.compareTo(targetUnit.scale)\r\n return when {\r\n

```

```

sourceCompareTarget > 0 -> value * (sourceUnit.scale / targetUnit.scale)\r\n sourceCompareTarget < 0 ->
value / (targetUnit.scale / sourceUnit.scale)\r\n else -> value\r\n }\r\n}\r\n\r\n", "/*\r\n * Copyright 2010-2019
JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.time\r\n\r\nimport
kotlin.js.json\r\nimport kotlin.math.*\r\n\r\ninternal actual fun formatToExactDecimals(value: Double, decimals:
Int): String {\r\n val rounded = if (decimals == 0) {\r\n value\r\n } else {\r\n val pow =
10.0.pow(decimals)\r\n @Suppress("\r\nDEPRECATION\r\n", "\r\nDEPRECATION_ERROR\r\n")\r\n
kotlin.js.Math.round(abs(value) * pow) / pow * sign(value)\r\n }\r\n return
rounded.asDynamic().toFixed(decimals).unsafeCast<String>()\r\n}\r\n\r\ninternal actual fun
formatUpToDecimals(value: Double, decimals: Int): String {\r\n return value.asDynamic().toLocaleString("\r\nen-
us\r\n", json("\r\nmaximumFractionDigits\r\n" to decimals)).unsafeCast<String>()\r\n}\r\n\r\ninternal actual fun
formatScientific(value: Double): String {\r\n return
value.asDynamic().toExponential(2).unsafeCast<String>()\r\n}\r\n", "/*\r\n * Copyright 2010-2019 JetBrains s.r.o. and
Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that
can be found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.time\r\n\r\nimport
org.w3c.performance.GlobalPerformance\r\n\r\nimport
org.w3c.performance.Performance\r\n\r\n@SinceKotlin("\r\n1.3\r\n")\r\n@ExperimentalTime\r\n\r\npublic actual object
MonoClock : Clock {\r\n\r\n private val actualClock: Clock = run {\r\n val isNode: Boolean = js("\r\ntypeof
process !== 'undefined' && process.versions && !process.versions.node\r\n")\r\n\r\n if (isNode)\r\n
HrTimeClock(js("\r\nprocess\r\n").unsafeCast<Process>())\r\n else\r\n
js("\r\nself\r\n").unsafeCast<GlobalPerformance?>()?performance?.let(::PerformanceClock)\r\n ?:
DateNowClock\r\n\r\n }\r\n\r\n override fun markNow(): ClockMark =
actualClock.markNow()\r\n}\r\n\r\ninternal external interface Process {\r\n fun hrtime(time: Array<Double> =
definedExternally): Array<Double>\r\n}\r\n\r\n@SinceKotlin("\r\n1.3\r\n")\r\n@ExperimentalTime\r\n\r\ninternal class
HrTimeClock(val process: Process) : Clock {\r\n\r\n override fun markNow(): ClockMark = object : ClockMark()
{\r\n val startedAt = process.hrtime()\r\n override fun elapsedNow(): Duration =\r\n
process.hrtime(startedAt).let { (seconds, nanos) -> seconds.seconds + nanos.nanoseconds }\r\n }\r\n\r\n override
fun toString(): String =
"\r\nClock(process.hrtime())\r\n"\r\n}\r\n\r\n@SinceKotlin("\r\n1.3\r\n")\r\n@ExperimentalTime\r\n\r\ninternal class
PerformanceClock(val performance: Performance) : AbstractDoubleClock(unit = DurationUnit.MILLISECONDS)
{\r\n override fun read(): Double = performance.now()\r\n override fun toString(): String =
"\r\nClock(self.performance.now())\r\n"\r\n}\r\n\r\n@SinceKotlin("\r\n1.3\r\n")\r\n@ExperimentalTime\r\n\r\ninternal object
DateNowClock : AbstractDoubleClock(unit = DurationUnit.MILLISECONDS) {\r\n override fun read(): Double
= kotlin.js.Date.now()\r\n override fun toString(): String = "\r\nClock(Date.now())\r\n", "/*\r\n * Copyright 2010-
2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by
the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n// NOTE: THIS FILE IS
AUTO-GENERATED, DO NOT EDIT!\r\n\r\n// See libraries/tools/idl2k for
details\r\n\r\n@file:Suppress("\r\nNESTED_CLASS_IN_EXTERNAL_INTERFACE\r\n")\r\n\r\npackage
org.khronos.webgl\r\n\r\nimport kotlin.js.*\r\nimport org.w3c.css.masking.*\r\nimport org.w3c.dom.*\r\nimport
org.w3c.dom.clipboard.*\r\nimport org.w3c.dom.css.*\r\nimport org.w3c.dom.events.*\r\nimport
org.w3c.dom.mediacapture.*\r\nimport org.w3c.dom.parsing.*\r\nimport org.w3c.dom.pointerevents.*\r\nimport
org.w3c.dom.svg.*\r\nimport org.w3c.dom.url.*\r\nimport org.w3c.fetch.*\r\nimport org.w3c.files.*\r\nimport
org.w3c.notifications.*\r\nimport org.w3c.performance.*\r\nimport org.w3c.workers.*\r\nimport
org.w3c.xhr.*\r\n\r\n\r\npublic external interface WebGLContextAttributes {\r\n var alpha: Boolean? /* = true */\r\n
 get() = definedExternally\r\n set(value) = definedExternally\r\n var depth: Boolean? /* = true */\r\n
 get() = definedExternally\r\n set(value) = definedExternally\r\n var stencil: Boolean? /* = false */\r\n
 get() = definedExternally\r\n set(value) = definedExternally\r\n var antialias: Boolean? /* = true */\r\n
 get() = definedExternally\r\n set(value) = definedExternally\r\n var premultipliedAlpha: Boolean? /* = true */\r\n

```

```

get() = definedExternally\r\n set(value) = definedExternally\r\n var preserveDrawingBuffer: Boolean? /* =
false */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var
preferLowPowerToHighPerformance: Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var failIfMajorPerformanceCaveat: Boolean? /* = false */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
WebGLContextAttributes(alpha: Boolean? = true, depth: Boolean? = true, stencil: Boolean? = false, antialias:
Boolean? = true, premultipliedAlpha: Boolean? = true, preserveDrawingBuffer: Boolean? = false,
preferLowPowerToHighPerformance: Boolean? = false, failIfMajorPerformanceCaveat: Boolean? = false):
WebGLContextAttributes {\r\n val o = js("{}")\r\n o["alpha"] = alpha\r\n o["depth"] = depth\r\n
o["stencil"] = stencil\r\n o["antialias"] = antialias\r\n o["premultipliedAlpha"] = premultipliedAlpha\r\n
o["preserveDrawingBuffer"] = preserveDrawingBuffer\r\n o["preferLowPowerToHighPerformance"] =
preferLowPowerToHighPerformance\r\n o["failIfMajorPerformanceCaveat"] =
failIfMajorPerformanceCaveat\r\n\r\n return o\r\n}\r\n\r\npublic external abstract class WebGLObject
{\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[WebGLBuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLBuffer) to Kotlin\r\n */\r\npublic external
abstract class WebGLBuffer : WebGLObject {\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[WebGLFramebuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLFramebuffer) to Kotlin\r\n
*/\r\npublic external abstract class WebGLFramebuffer : WebGLObject {\r\n}\r\n\r\n/**\r\n * Exposes the
JavaScript [WebGLProgram](https://developer.mozilla.org/en/docs/Web/API/WebGLProgram) to Kotlin\r\n
*/\r\npublic external abstract class WebGLProgram : WebGLObject {\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[WebGLRenderbuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLRenderbuffer) to Kotlin\r\n
*/\r\npublic external abstract class WebGLRenderbuffer : WebGLObject {\r\n}\r\n\r\n/**\r\n * Exposes the
JavaScript [WebGLShader](https://developer.mozilla.org/en/docs/Web/API/WebGLShader) to Kotlin\r\n
*/\r\npublic external abstract class WebGLShader : WebGLObject {\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[WebGLTexture](https://developer.mozilla.org/en/docs/Web/API/WebGLTexture) to Kotlin\r\n */\r\npublic
external abstract class WebGLTexture : WebGLObject {\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[WebGLUniformLocation](https://developer.mozilla.org/en/docs/Web/API/WebGLUniformLocation) to Kotlin\r\n
*/\r\npublic external abstract class WebGLUniformLocation {\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[WebGLActiveInfo](https://developer.mozilla.org/en/docs/Web/API/WebGLActiveInfo) to Kotlin\r\n */\r\npublic
external abstract class WebGLActiveInfo {\r\n open val size: Int\r\n open val type: Int\r\n open val name:
String\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[WebGLShaderPrecisionFormat](https://developer.mozilla.org/en/docs/Web/API/WebGLShaderPrecisionFormat) to
Kotlin\r\n */\r\npublic external abstract class WebGLShaderPrecisionFormat {\r\n open val rangeMin: Int\r\n
open val rangeMax: Int\r\n open val precision: Int\r\n}\r\n\r\npublic external interface
WebGLRenderingContextBase {\r\n val canvas: HTMLCanvasElement\r\n val drawingBufferWidth: Int\r\n
val drawingBufferHeight: Int\r\n fun getContextAttributes(): WebGLContextAttributes?\r\n fun isContextLost():
Boolean\r\n fun getSupportedExtensions(): Array<String>?\r\n fun getExtension(name: String): dynamic\r\n
fun activeTexture(texture: Int): Unit\r\n fun attachShader(program: WebGLProgram?, shader: WebGLShader?):
Unit\r\n fun bindAttribLocation(program: WebGLProgram?, index: Int, name: String): Unit\r\n fun
bindBuffer(target: Int, buffer: WebGLBuffer?): Unit\r\n fun bindFramebuffer(target: Int, framebuffer:
WebGLFramebuffer?): Unit\r\n fun bindRenderbuffer(target: Int, renderbuffer: WebGLRenderbuffer?): Unit\r\n
fun bindTexture(target: Int, texture: WebGLTexture?): Unit\r\n fun blendColor(red: Float, green: Float, blue:
Float, alpha: Float): Unit\r\n fun blendEquation(mode: Int): Unit\r\n fun blendEquationSeparate(modeRGB: Int,
modeAlpha: Int): Unit\r\n fun blendFunc(sfactor: Int, dfactor: Int): Unit\r\n fun blendFuncSeparate(srcRGB: Int,
dstRGB: Int, srcAlpha: Int, dstAlpha: Int): Unit\r\n fun bufferData(target: Int, size: Int, usage: Int): Unit\r\n
fun bufferData(target: Int, data: BufferDataSource?, usage: Int): Unit\r\n fun bufferSubData(target: Int, offset: Int,
data: BufferDataSource?): Unit\r\n fun checkFramebufferStatus(target: Int): Int\r\n fun clear(mask: Int): Unit\r\n
fun clearColor(red: Float, green: Float, blue: Float, alpha: Float): Unit\r\n fun clearDepth(depth: Float): Unit\r\n

```

fun clearStencil(s: Int): Unit\r\n fun colorMask(red: Boolean, green: Boolean, blue: Boolean, alpha: Boolean):  
 Unit\r\n fun compileShader(shader: WebGLShader?): Unit\r\n fun compressedTexImage2D(target: Int, level:  
 Int, internalformat: Int, width: Int, height: Int, border: Int, data: ArrayBufferView): Unit\r\n fun  
 compressedTexSubImage2D(target: Int, level: Int, xoffset: Int, yoffset: Int, width: Int, height: Int, format: Int, data:  
 ArrayBufferView): Unit\r\n fun copyTexImage2D(target: Int, level: Int, internalformat: Int, x: Int, y: Int, width:  
 Int, height: Int, border: Int): Unit\r\n fun copyTexSubImage2D(target: Int, level: Int, xoffset: Int, yoffset: Int, x:  
 Int, y: Int, width: Int, height: Int): Unit\r\n fun createBuffer(): WebGLBuffer?\r\n fun createFramebuffer():  
 WebGLFramebuffer?\r\n fun createProgram(): WebGLProgram?\r\n fun createRenderbuffer():  
 WebGLRenderbuffer?\r\n fun createShader(type: Int): WebGLShader?\r\n fun createTexture():  
 WebGLTexture?\r\n fun cullFace(mode: Int): Unit\r\n fun deleteBuffer(buffer: WebGLBuffer?): Unit\r\n fun  
 deleteFramebuffer(framebuffer: WebGLFramebuffer?): Unit\r\n fun deleteProgram(program: WebGLProgram?):  
 Unit\r\n fun deleteRenderbuffer(renderbuffer: WebGLRenderbuffer?): Unit\r\n fun deleteShader(shader:  
 WebGLShader?): Unit\r\n fun deleteTexture(texture: WebGLTexture?): Unit\r\n fun depthFunc(func: Int):  
 Unit\r\n fun depthMask(flag: Boolean): Unit\r\n fun depthRange(zNear: Float, zFar: Float): Unit\r\n fun  
 detachShader(program: WebGLProgram?, shader: WebGLShader?): Unit\r\n fun disable(cap: Int): Unit\r\n fun  
 disableVertexArray(index: Int): Unit\r\n fun drawArrays(mode: Int, first: Int, count: Int): Unit\r\n fun  
 drawElements(mode: Int, count: Int, type: Int, offset: Int): Unit\r\n fun enable(cap: Int): Unit\r\n fun  
 enableVertexArray(index: Int): Unit\r\n fun finish(): Unit\r\n fun flush(): Unit\r\n fun  
 framebufferRenderbuffer(target: Int, attachment: Int, renderbuffertarget: Int, renderbuffer: WebGLRenderbuffer?):  
 Unit\r\n fun framebufferTexture2D(target: Int, attachment: Int, textarget: Int, texture: WebGLTexture?, level: Int):  
 Unit\r\n fun frontFace(mode: Int): Unit\r\n fun generateMipmap(target: Int): Unit\r\n fun  
 getActiveAttrib(program: WebGLProgram?, index: Int): WebGLActiveInfo?\r\n fun getActiveUniform(program:  
 WebGLProgram?, index: Int): WebGLActiveInfo?\r\n fun getAttachedShaders(program: WebGLProgram?):  
 Array<WebGLShader?>\r\n fun getAttribLocation(program: WebGLProgram?, name: String): Int\r\n fun  
 getBufferParameter(target: Int, pname: Int): Any?\r\n fun getParameter(pname: Int): Any?\r\n fun getError():  
 Int\r\n fun getFramebufferAttachmentParameter(target: Int, attachment: Int, pname: Int): Any?\r\n fun  
 getProgramParameter(program: WebGLProgram?, pname: Int): Any?\r\n fun getProgramInfoLog(program:  
 WebGLProgram?): String?\r\n fun getRenderbufferParameter(target: Int, pname: Int): Any?\r\n fun  
 getShaderParameter(shader: WebGLShader?, pname: Int): Any?\r\n fun getShaderPrecisionFormat(shaderType:  
 Int, precisionType: Int): WebGLShaderPrecisionFormat?\r\n fun getShaderInfoLog(shader: WebGLShader?):  
 String?\r\n fun getShaderSource(shader: WebGLShader?): String?\r\n fun getTexParameter(target: Int, pname:  
 Int): Any?\r\n fun getUniform(program: WebGLProgram?, location: WebGLUniformLocation?): Any?\r\n fun  
 glGetUniformLocation(program: WebGLProgram?, name: String): WebGLUniformLocation?\r\n fun  
 glVertexAttrib(index: Int, pname: Int): Any?\r\n fun glVertexAttribOffset(index: Int, pname: Int): Int\r\n fun  
 hint(target: Int, mode: Int): Unit\r\n fun isBuffer(buffer: WebGLBuffer?): Boolean\r\n fun isEnabled(cap: Int):  
 Boolean\r\n fun isFramebuffer(framebuffer: WebGLFramebuffer?): Boolean\r\n fun isProgram(program:  
 WebGLProgram?): Boolean\r\n fun isRenderbuffer(renderbuffer: WebGLRenderbuffer?): Boolean\r\n fun  
 isShader(shader: WebGLShader?): Boolean\r\n fun isTexture(texture: WebGLTexture?): Boolean\r\n fun  
 lineWidth(width: Float): Unit\r\n fun linkProgram(program: WebGLProgram?): Unit\r\n fun pixelStorei(pname:  
 Int, param: Int): Unit\r\n fun polygonOffset(factor: Float, units: Float): Unit\r\n fun readPixels(x: Int, y: Int,  
 width: Int, height: Int, format: Int, type: Int, pixels: ArrayBufferView?): Unit\r\n fun renderbufferStorage(target:  
 Int, internalformat: Int, width: Int, height: Int): Unit\r\n fun sampleCoverage(value: Float, invert: Boolean):  
 Unit\r\n fun scissor(x: Int, y: Int, width: Int, height: Int): Unit\r\n fun shaderSource(shader: WebGLShader?,  
 source: String): Unit\r\n fun stencilFunc(func: Int, ref: Int, mask: Int): Unit\r\n fun stencilFuncSeparate(face: Int,  
 func: Int, ref: Int, mask: Int): Unit\r\n fun stencilMask(mask: Int): Unit\r\n fun stencilMaskSeparate(face: Int,  
 mask: Int): Unit\r\n fun stencilOp(fail: Int, zfail: Int, zpass: Int): Unit\r\n fun stencilOpSeparate(face: Int, fail:  
 Int, zfail: Int, zpass: Int): Unit\r\n fun texImage2D(target: Int, level: Int, internalformat: Int, width: Int, height: Int,  
 border: Int, format: Int, type: Int, pixels: ArrayBufferView?): Unit\r\n fun texImage2D(target: Int, level: Int,

```

internalformat: Int, format: Int, type: Int, source: TexImageSource?): Unit\r\n fun texParameterf(target: Int,
pname: Int, param: Float): Unit\r\n fun texParameteri(target: Int, pname: Int, param: Int): Unit\r\n fun
texSubImage2D(target: Int, level: Int, xoffset: Int, yoffset: Int, width: Int, height: Int, format: Int, type: Int, pixels:
ArrayBufferView?): Unit\r\n fun texSubImage2D(target: Int, level: Int, xoffset: Int, yoffset: Int, format: Int, type:
Int, source: TexImageSource?): Unit\r\n fun uniform1f(location: WebGLUniformLocation?, x: Float): Unit\r\n
fun uniform1fv(location: WebGLUniformLocation?, v: Float32Array): Unit\r\n fun uniform1fv(location:
WebGLUniformLocation?, v: Array<Float>): Unit\r\n fun uniform1i(location: WebGLUniformLocation?, x: Int):
Unit\r\n fun uniform1iv(location: WebGLUniformLocation?, v: Int32Array): Unit\r\n fun uniform1iv(location:
WebGLUniformLocation?, v: Array<Int>): Unit\r\n fun uniform2f(location: WebGLUniformLocation?, x: Float,
y: Float): Unit\r\n fun uniform2fv(location: WebGLUniformLocation?, v: Float32Array): Unit\r\n fun
uniform2fv(location: WebGLUniformLocation?, v: Array<Float>): Unit\r\n fun uniform2i(location:
WebGLUniformLocation?, x: Int, y: Int): Unit\r\n fun uniform2iv(location: WebGLUniformLocation?, v:
Int32Array): Unit\r\n fun uniform2iv(location: WebGLUniformLocation?, v: Array<Int>): Unit\r\n fun
uniform3f(location: WebGLUniformLocation?, x: Float, y: Float, z: Float): Unit\r\n fun uniform3fv(location:
WebGLUniformLocation?, v: Float32Array): Unit\r\n fun uniform3fv(location: WebGLUniformLocation?, v:
Array<Float>): Unit\r\n fun uniform3i(location: WebGLUniformLocation?, x: Int, y: Int, z: Int): Unit\r\n fun
uniform3iv(location: WebGLUniformLocation?, v: Int32Array): Unit\r\n fun uniform3iv(location:
WebGLUniformLocation?, v: Array<Int>): Unit\r\n fun uniform4f(location: WebGLUniformLocation?, x: Float,
y: Float, z: Float, w: Float): Unit\r\n fun uniform4fv(location: WebGLUniformLocation?, v: Float32Array):
Unit\r\n fun uniform4fv(location: WebGLUniformLocation?, v: Array<Float>): Unit\r\n fun uniform4i(location:
WebGLUniformLocation?, x: Int, y: Int, z: Int, w: Int): Unit\r\n fun uniform4iv(location:
WebGLUniformLocation?, v: Int32Array): Unit\r\n fun uniform4iv(location: WebGLUniformLocation?, v:
Array<Int>): Unit\r\n fun uniformMatrix2fv(location: WebGLUniformLocation?, transpose: Boolean, value:
Float32Array): Unit\r\n fun uniformMatrix2fv(location: WebGLUniformLocation?, transpose: Boolean, value:
Array<Float>): Unit\r\n fun uniformMatrix3fv(location: WebGLUniformLocation?, transpose: Boolean, value:
Float32Array): Unit\r\n fun uniformMatrix3fv(location: WebGLUniformLocation?, transpose: Boolean, value:
Array<Float>): Unit\r\n fun uniformMatrix4fv(location: WebGLUniformLocation?, transpose: Boolean, value:
Float32Array): Unit\r\n fun uniformMatrix4fv(location: WebGLUniformLocation?, transpose: Boolean, value:
Array<Float>): Unit\r\n fun useProgram(program: WebGLProgram?): Unit\r\n fun validateProgram(program:
WebGLProgram?): Unit\r\n fun vertexAttrib1f(index: Int, x: Float): Unit\r\n fun vertexAttrib1fv(index: Int,
values: dynamic): Unit\r\n fun vertexAttrib2f(index: Int, x: Float, y: Float): Unit\r\n fun vertexAttrib2fv(index:
Int, values: dynamic): Unit\r\n fun vertexAttrib3f(index: Int, x: Float, y: Float, z: Float): Unit\r\n fun
vertexAttrib3fv(index: Int, values: dynamic): Unit\r\n fun vertexAttrib4f(index: Int, x: Float, y: Float, z: Float, w:
Float): Unit\r\n fun vertexAttrib4fv(index: Int, values: dynamic): Unit\r\n fun vertexAttribPointer(index: Int,
size: Int, type: Int, normalized: Boolean, stride: Int, offset: Int): Unit\r\n fun viewport(x: Int, y: Int, width: Int,
height: Int): Unit\r\n\r\n companion object {\r\n val DEPTH_BUFFER_BIT: Int\r\n val
STENCIL_BUFFER_BIT: Int\r\n val COLOR_BUFFER_BIT: Int\r\n val POINTS: Int\r\n val LINES:
Int\r\n val LINE_LOOP: Int\r\n val LINE_STRIP: Int\r\n val TRIANGLES: Int\r\n val
TRIANGLE_STRIP: Int\r\n val TRIANGLE_FAN: Int\r\n val ZERO: Int\r\n val ONE: Int\r\n val
SRC_COLOR: Int\r\n val ONE_MINUS_SRC_COLOR: Int\r\n val SRC_ALPHA: Int\r\n val
ONE_MINUS_SRC_ALPHA: Int\r\n val DST_ALPHA: Int\r\n val ONE_MINUS_DST_ALPHA: Int\r\n
val DST_COLOR: Int\r\n val ONE_MINUS_DST_COLOR: Int\r\n val SRC_ALPHA_SATURATE:
Int\r\n val FUNC_ADD: Int\r\n val BLEND_EQUATION: Int\r\n val BLEND_EQUATION_RGB:
Int\r\n val BLEND_EQUATION_ALPHA: Int\r\n val FUNC_SUBTRACT: Int\r\n val
FUNC_REVERSE_SUBTRACT: Int\r\n val BLEND_DST_RGB: Int\r\n val BLEND_SRC_RGB: Int\r\n
val BLEND_DST_ALPHA: Int\r\n val BLEND_SRC_ALPHA: Int\r\n val CONSTANT_COLOR: Int\r\n
val ONE_MINUS_CONSTANT_COLOR: Int\r\n val CONSTANT_ALPHA: Int\r\n val
ONE_MINUS_CONSTANT_ALPHA: Int\r\n val BLEND_COLOR: Int\r\n val ARRAY_BUFFER: Int\r\n

```

val ELEMENT\_ARRAY\_BUFFER: Int\r\n     val ARRAY\_BUFFER\_BINDING: Int\r\n     val  
 ELEMENT\_ARRAY\_BUFFER\_BINDING: Int\r\n     val STREAM\_DRAW: Int\r\n     val STATIC\_DRAW:  
 Int\r\n     val DYNAMIC\_DRAW: Int\r\n     val BUFFER\_SIZE: Int\r\n     val BUFFER\_USAGE: Int\r\n  
 val CURRENT\_VERTEX\_ATTRIB: Int\r\n     val FRONT: Int\r\n     val BACK: Int\r\n     val  
 FRONT\_AND\_BACK: Int\r\n     val CULL\_FACE: Int\r\n     val BLEND: Int\r\n     val DITHER: Int\r\n  
 val STENCIL\_TEST: Int\r\n     val DEPTH\_TEST: Int\r\n     val SCISSOR\_TEST: Int\r\n     val  
 POLYGON\_OFFSET\_FILL: Int\r\n     val SAMPLE\_ALPHA\_TO\_COVERAGE: Int\r\n     val  
 SAMPLE\_COVERAGE: Int\r\n     val NO\_ERROR: Int\r\n     val INVALID\_ENUM: Int\r\n     val  
 INVALID\_VALUE: Int\r\n     val INVALID\_OPERATION: Int\r\n     val OUT\_OF\_MEMORY: Int\r\n     val  
 CW: Int\r\n     val CCW: Int\r\n     val LINE\_WIDTH: Int\r\n     val ALIASED\_POINT\_SIZE\_RANGE:  
 Int\r\n     val ALIASED\_LINE\_WIDTH\_RANGE: Int\r\n     val CULL\_FACE\_MODE: Int\r\n     val  
 FRONT\_FACE: Int\r\n     val DEPTH\_RANGE: Int\r\n     val DEPTH\_WRITEMASK: Int\r\n     val  
 DEPTH\_CLEAR\_VALUE: Int\r\n     val DEPTH\_FUNC: Int\r\n     val STENCIL\_CLEAR\_VALUE: Int\r\n  
 val STENCIL\_FUNC: Int\r\n     val STENCIL\_FAIL: Int\r\n     val STENCIL\_PASS\_DEPTH\_FAIL: Int\r\n  
 val STENCIL\_PASS\_DEPTH\_PASS: Int\r\n     val STENCIL\_REF: Int\r\n     val STENCIL\_VALUE\_MASK:  
 Int\r\n     val STENCIL\_WRITEMASK: Int\r\n     val STENCIL\_BACK\_FUNC: Int\r\n     val  
 STENCIL\_BACK\_FAIL: Int\r\n     val STENCIL\_BACK\_PASS\_DEPTH\_FAIL: Int\r\n     val  
 STENCIL\_BACK\_PASS\_DEPTH\_PASS: Int\r\n     val STENCIL\_BACK\_REF: Int\r\n     val  
 STENCIL\_BACK\_VALUE\_MASK: Int\r\n     val STENCIL\_BACK\_WRITEMASK: Int\r\n     val  
 VIEWPORT: Int\r\n     val SCISSOR\_BOX: Int\r\n     val COLOR\_CLEAR\_VALUE: Int\r\n     val  
 COLOR\_WRITEMASK: Int\r\n     val UNPACK\_ALIGNMENT: Int\r\n     val PACK\_ALIGNMENT: Int\r\n  
 val MAX\_TEXTURE\_SIZE: Int\r\n     val MAX\_VIEWPORT\_DIMS: Int\r\n     val SUBPIXEL\_BITS: Int\r\n  
 val RED\_BITS: Int\r\n     val GREEN\_BITS: Int\r\n     val BLUE\_BITS: Int\r\n     val ALPHA\_BITS:  
 Int\r\n     val DEPTH\_BITS: Int\r\n     val STENCIL\_BITS: Int\r\n     val POLYGON\_OFFSET\_UNITS:  
 Int\r\n     val POLYGON\_OFFSET\_FACTOR: Int\r\n     val TEXTURE\_BINDING\_2D: Int\r\n     val  
 SAMPLE\_BUFFERS: Int\r\n     val SAMPLES: Int\r\n     val SAMPLE\_COVERAGE\_VALUE: Int\r\n     val  
 SAMPLE\_COVERAGE\_INVERT: Int\r\n     val COMPRESSED\_TEXTURE\_FORMATS: Int\r\n     val  
 DONT\_CARE: Int\r\n     val FASTEST: Int\r\n     val NICEST: Int\r\n     val GENERATE\_MIPMAP\_HINT:  
 Int\r\n     val BYTE: Int\r\n     val UNSIGNED\_BYTE: Int\r\n     val SHORT: Int\r\n     val  
 UNSIGNED\_SHORT: Int\r\n     val INT: Int\r\n     val UNSIGNED\_INT: Int\r\n     val FLOAT: Int\r\n  
 val DEPTH\_COMPONENT: Int\r\n     val ALPHA: Int\r\n     val RGB: Int\r\n     val RGBA: Int\r\n     val  
 LUMINANCE: Int\r\n     val LUMINANCE\_ALPHA: Int\r\n     val UNSIGNED\_SHORT\_4\_4\_4\_4: Int\r\n  
 val UNSIGNED\_SHORT\_5\_5\_5\_1: Int\r\n     val UNSIGNED\_SHORT\_5\_6\_5: Int\r\n     val  
 FRAGMENT\_SHADER: Int\r\n     val VERTEX\_SHADER: Int\r\n     val MAX\_VERTEX\_ATTRIBS: Int\r\n  
 val MAX\_VERTEX\_UNIFORM\_VECTORS: Int\r\n     val MAX\_VARYING\_VECTORS: Int\r\n     val  
 MAX\_COMBINED\_TEXTURE\_IMAGE\_UNITS: Int\r\n     val MAX\_VERTEX\_TEXTURE\_IMAGE\_UNITS:  
 Int\r\n     val MAX\_TEXTURE\_IMAGE\_UNITS: Int\r\n     val MAX\_FRAGMENT\_UNIFORM\_VECTORS:  
 Int\r\n     val SHADER\_TYPE: Int\r\n     val DELETE\_STATUS: Int\r\n     val LINK\_STATUS: Int\r\n  
 val VALIDATE\_STATUS: Int\r\n     val ATTACHED\_SHADERS: Int\r\n     val ACTIVE\_UNIFORMS: Int\r\n  
 val ACTIVE\_ATTRIBUTES: Int\r\n     val SHADING\_LANGUAGE\_VERSION: Int\r\n     val  
 CURRENT\_PROGRAM: Int\r\n     val NEVER: Int\r\n     val LESS: Int\r\n     val EQUAL: Int\r\n     val  
 LEQUAL: Int\r\n     val GREATER: Int\r\n     val NOTEQUAL: Int\r\n     val GEQUAL: Int\r\n     val  
 ALWAYS: Int\r\n     val KEEP: Int\r\n     val REPLACE: Int\r\n     val INCR: Int\r\n     val DECR: Int\r\n  
 val INVERT: Int\r\n     val INCR\_WRAP: Int\r\n     val DECR\_WRAP: Int\r\n     val VENDOR: Int\r\n  
 val RENDERER: Int\r\n     val VERSION: Int\r\n     val NEAREST: Int\r\n     val LINEAR: Int\r\n     val  
 NEAREST\_MIPMAP\_NEAREST: Int\r\n     val LINEAR\_MIPMAP\_NEAREST: Int\r\n     val  
 NEAREST\_MIPMAP\_LINEAR: Int\r\n     val LINEAR\_MIPMAP\_LINEAR: Int\r\n     val  
 TEXTURE\_MAG\_FILTER: Int\r\n     val TEXTURE\_MIN\_FILTER: Int\r\n     val TEXTURE\_WRAP\_S:

```

Int\r\n val TEXTURE_WRAP_T: Int\r\n val TEXTURE_2D: Int\r\n val TEXTURE: Int\r\n val
TEXTURE_CUBE_MAP: Int\r\n val TEXTURE_BINDING_CUBE_MAP: Int\r\n val
TEXTURE_CUBE_MAP_POSITIVE_X: Int\r\n val TEXTURE_CUBE_MAP_NEGATIVE_X: Int\r\n val
TEXTURE_CUBE_MAP_POSITIVE_Y: Int\r\n val TEXTURE_CUBE_MAP_NEGATIVE_Y: Int\r\n val
TEXTURE_CUBE_MAP_POSITIVE_Z: Int\r\n val TEXTURE_CUBE_MAP_NEGATIVE_Z: Int\r\n val
MAX_CUBE_MAP_TEXTURE_SIZE: Int\r\n val TEXTURE0: Int\r\n val TEXTURE1: Int\r\n val
TEXTURE2: Int\r\n val TEXTURE3: Int\r\n val TEXTURE4: Int\r\n val TEXTURE5: Int\r\n val
TEXTURE6: Int\r\n val TEXTURE7: Int\r\n val TEXTURE8: Int\r\n val TEXTURE9: Int\r\n val
TEXTURE10: Int\r\n val TEXTURE11: Int\r\n val TEXTURE12: Int\r\n val TEXTURE13: Int\r\n
val TEXTURE14: Int\r\n val TEXTURE15: Int\r\n val TEXTURE16: Int\r\n val TEXTURE17: Int\r\n
 val TEXTURE18: Int\r\n val TEXTURE19: Int\r\n val TEXTURE20: Int\r\n val TEXTURE21:
Int\r\n val TEXTURE22: Int\r\n val TEXTURE23: Int\r\n val TEXTURE24: Int\r\n val
TEXTURE25: Int\r\n val TEXTURE26: Int\r\n val TEXTURE27: Int\r\n val TEXTURE28: Int\r\n
val TEXTURE29: Int\r\n val TEXTURE30: Int\r\n val TEXTURE31: Int\r\n val ACTIVE_TEXTURE:
Int\r\n val REPEAT: Int\r\n val CLAMP_TO_EDGE: Int\r\n val MIRRORED_REPEAT: Int\r\n
val FLOAT_VEC2: Int\r\n val FLOAT_VEC3: Int\r\n val FLOAT_VEC4: Int\r\n val INT_VEC2:
Int\r\n val INT_VEC3: Int\r\n val INT_VEC4: Int\r\n val BOOL: Int\r\n val BOOL_VEC2: Int\r\n
 val BOOL_VEC3: Int\r\n val BOOL_VEC4: Int\r\n val FLOAT_MAT2: Int\r\n val FLOAT_MAT3:
Int\r\n val FLOAT_MAT4: Int\r\n val SAMPLER_2D: Int\r\n val SAMPLER_CUBE: Int\r\n val
VERTEX_ATTRIB_ARRAY_ENABLED: Int\r\n val VERTEX_ATTRIB_ARRAY_SIZE: Int\r\n val
VERTEX_ATTRIB_ARRAY_STRIDE: Int\r\n val VERTEX_ATTRIB_ARRAY_TYPE: Int\r\n val
VERTEX_ATTRIB_ARRAY_NORMALIZED: Int\r\n val VERTEX_ATTRIB_ARRAY_POINTER: Int\r\n
val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int\r\n val
IMPLEMENTATION_COLOR_READ_TYPE: Int\r\n val IMPLEMENTATION_COLOR_READ_FORMAT:
Int\r\n val COMPILE_STATUS: Int\r\n val LOW_FLOAT: Int\r\n val MEDIUM_FLOAT: Int\r\n
val HIGH_FLOAT: Int\r\n val LOW_INT: Int\r\n val MEDIUM_INT: Int\r\n val HIGH_INT: Int\r\n
 val FRAMEBUFFER: Int\r\n val RENDERBUFFER: Int\r\n val RGBA4: Int\r\n val RGB5_A1:
Int\r\n val RGB565: Int\r\n val DEPTH_COMPONENT16: Int\r\n val STENCIL_INDEX: Int\r\n
val STENCIL_INDEX8: Int\r\n val DEPTH_STENCIL: Int\r\n val RENDERBUFFER_WIDTH: Int\r\n
val RENDERBUFFER_HEIGHT: Int\r\n val RENDERBUFFER_INTERNAL_FORMAT: Int\r\n val
RENDERBUFFER_RED_SIZE: Int\r\n val RENDERBUFFER_GREEN_SIZE: Int\r\n val
RENDERBUFFER_BLUE_SIZE: Int\r\n val RENDERBUFFER_ALPHA_SIZE: Int\r\n val
RENDERBUFFER_DEPTH_SIZE: Int\r\n val RENDERBUFFER_STENCIL_SIZE: Int\r\n val
FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int\r\n val
FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int\r\n val
FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int\r\n val
FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int\r\n val COLOR_ATTACHMENT0:
Int\r\n val DEPTH_ATTACHMENT: Int\r\n val STENCIL_ATTACHMENT: Int\r\n val
DEPTH_STENCIL_ATTACHMENT: Int\r\n val NONE: Int\r\n val FRAMEBUFFER_COMPLETE:
Int\r\n val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int\r\n val
FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int\r\n val
FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int\r\n val FRAMEBUFFER_UNSUPPORTED: Int\r\n
val FRAMEBUFFER_BINDING: Int\r\n val RENDERBUFFER_BINDING: Int\r\n val
MAX_RENDERBUFFER_SIZE: Int\r\n val INVALID_FRAMEBUFFER_OPERATION: Int\r\n val
UNPACK_FLIP_Y_WEBGL: Int\r\n val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int\r\n val
CONTEXT_LOST_WEBGL: Int\r\n val UNPACK_COLORSPACE_CONVERSION_WEBGL: Int\r\n val
BROWSER_DEFAULT_WEBGL: Int\r\n }\r\n}\r\n\r\n**\r\n * Exposes the JavaScript
[WebGLRenderingContext](https://developer.mozilla.org/en/docs/Web/API/WebGLRenderingContext) to

```



```

Kotlin\n */\npublic external abstract class WebGLRenderingContext : WebGLRenderingContextBase,
RenderingContext {\n\n companion object {\n val DEPTH_BUFFER_BIT: Int\n val STENCIL_BUFFER_BIT: Int\n val COLOR_BUFFER_BIT: Int\n val POINTS: Int\n val LINES: Int\n val LINE_LOOP: Int\n val LINE_STRIP: Int\n val TRIANGLES: Int\n val TRIANGLE_STRIP: Int\n val TRIANGLE_FAN: Int\n val ZERO: Int\n val ONE: Int\n val SRC_COLOR: Int\n val ONE_MINUS_SRC_COLOR: Int\n val SRC_ALPHA: Int\n val ONE_MINUS_SRC_ALPHA: Int\n val DST_ALPHA: Int\n val ONE_MINUS_DST_ALPHA: Int\n val DST_COLOR: Int\n val ONE_MINUS_DST_COLOR: Int\n val SRC_ALPHA_SATURATE: Int\n val FUNC_ADD: Int\n val BLEND_EQUATION: Int\n val BLEND_EQUATION_RGB: Int\n val BLEND_EQUATION_ALPHA: Int\n val FUNC_SUBTRACT: Int\n val FUNC_REVERSE_SUBTRACT: Int\n val BLEND_DST_RGB: Int\n val BLEND_SRC_RGB: Int\n val BLEND_DST_ALPHA: Int\n val BLEND_SRC_ALPHA: Int\n val CONSTANT_COLOR: Int\n val ONE_MINUS_CONSTANT_COLOR: Int\n val CONSTANT_ALPHA: Int\n val ONE_MINUS_CONSTANT_ALPHA: Int\n val BLEND_COLOR: Int\n val ARRAY_BUFFER: Int\n val ELEMENT_ARRAY_BUFFER: Int\n val ARRAY_BUFFER_BINDING: Int\n val ELEMENT_ARRAY_BUFFER_BINDING: Int\n val STREAM_DRAW: Int\n val STATIC_DRAW: Int\n val DYNAMIC_DRAW: Int\n val BUFFER_SIZE: Int\n val BUFFER_USAGE: Int\n val CURRENT_VERTEX_ATTRIB: Int\n val FRONT: Int\n val BACK: Int\n val FRONT_AND_BACK: Int\n val CULL_FACE: Int\n val BLEND: Int\n val DITHER: Int\n val STENCIL_TEST: Int\n val DEPTH_TEST: Int\n val SCISSOR_TEST: Int\n val POLYGON_OFFSET_FILL: Int\n val SAMPLE_ALPHA_TO_COVERAGE: Int\n val SAMPLE_COVERAGE: Int\n val NO_ERROR: Int\n val INVALID_ENUM: Int\n val INVALID_VALUE: Int\n val INVALID_OPERATION: Int\n val OUT_OF_MEMORY: Int\n val CW: Int\n val CCW: Int\n val LINE_WIDTH: Int\n val ALIASED_POINT_SIZE_RANGE: Int\n val ALIASED_LINE_WIDTH_RANGE: Int\n val CULL_FACE_MODE: Int\n val FRONT_FACE: Int\n val DEPTH_RANGE: Int\n val DEPTH_WRITEMASK: Int\n val DEPTH_CLEAR_VALUE: Int\n val DEPTH_FUNC: Int\n val STENCIL_CLEAR_VALUE: Int\n val STENCIL_FUNC: Int\n val STENCIL_FAIL: Int\n val STENCIL_PASS_DEPTH_FAIL: Int\n val STENCIL_PASS_DEPTH_PASS: Int\n val STENCIL_REF: Int\n val STENCIL_VALUE_MASK: Int\n val STENCIL_WRITEMASK: Int\n val STENCIL_BACK_FUNC: Int\n val STENCIL_BACK_FAIL: Int\n val STENCIL_BACK_PASS_DEPTH_FAIL: Int\n val STENCIL_BACK_PASS_DEPTH_PASS: Int\n val STENCIL_BACK_REF: Int\n val STENCIL_BACK_VALUE_MASK: Int\n val STENCIL_BACK_WRITEMASK: Int\n val VIEWPORT: Int\n val SCISSOR_BOX: Int\n val COLOR_CLEAR_VALUE: Int\n val COLOR_WRITEMASK: Int\n val UNPACK_ALIGNMENT: Int\n val PACK_ALIGNMENT: Int\n val MAX_TEXTURE_SIZE: Int\n val MAX_VIEWPORT_DIMS: Int\n val SUBPIXEL_BITS: Int\n val RED_BITS: Int\n val GREEN_BITS: Int\n val BLUE_BITS: Int\n val ALPHA_BITS: Int\n val DEPTH_BITS: Int\n val STENCIL_BITS: Int\n val POLYGON_OFFSET_UNITS: Int\n val POLYGON_OFFSET_FACTOR: Int\n val TEXTURE_BINDING_2D: Int\n val SAMPLE_BUFFERS: Int\n val SAMPLES: Int\n val SAMPLE_COVERAGE_VALUE: Int\n val SAMPLE_COVERAGE_INVERT: Int\n val COMPRESSED_TEXTURE_FORMATS: Int\n val DONT_CARE: Int\n val FASTEST: Int\n val NICEST: Int\n val GENERATE_MIPMAP_HINT: Int\n val BYTE: Int\n val UNSIGNED_BYTE: Int\n val SHORT: Int\n val UNSIGNED_SHORT: Int\n val INT: Int\n val UNSIGNED_INT: Int\n val FLOAT: Int\n val DEPTH_COMPONENT: Int\n val ALPHA: Int\n val RGB: Int\n val RGBA: Int\n val LUMINANCE: Int\n val LUMINANCE_ALPHA: Int\n val UNSIGNED_SHORT_4_4_4_4: Int\n val UNSIGNED_SHORT_5_5_5_1: Int\n val UNSIGNED_SHORT_5_6_5: Int\n val FRAGMENT_SHADER: Int\n val VERTEX_SHADER: Int\n val MAX_VERTEX_ATTRIBS: Int

```

val MAX\_VERTEX\_UNIFORM\_VECTORS: Int\r\n     val MAX\_VARYING\_VECTORS: Int\r\n     val  
 MAX\_COMBINED\_TEXTURE\_IMAGE\_UNITS: Int\r\n     val MAX\_VERTEX\_TEXTURE\_IMAGE\_UNITS:  
 Int\r\n     val MAX\_TEXTURE\_IMAGE\_UNITS: Int\r\n     val MAX\_FRAGMENT\_UNIFORM\_VECTORS:  
 Int\r\n     val SHADER\_TYPE: Int\r\n     val DELETE\_STATUS: Int\r\n     val LINK\_STATUS: Int\r\n  
 val VALIDATE\_STATUS: Int\r\n     val ATTACHED\_SHADERS: Int\r\n     val ACTIVE\_UNIFORMS: Int\r\n  
    val ACTIVE\_ATTRIBUTES: Int\r\n     val SHADING\_LANGUAGE\_VERSION: Int\r\n     val  
 CURRENT\_PROGRAM: Int\r\n     val NEVER: Int\r\n     val LESS: Int\r\n     val EQUAL: Int\r\n     val  
 LEQUAL: Int\r\n     val GREATER: Int\r\n     val NOTEQUAL: Int\r\n     val GEQUAL: Int\r\n     val  
 ALWAYS: Int\r\n     val KEEP: Int\r\n     val REPLACE: Int\r\n     val INCR: Int\r\n     val DECR: Int\r\n  
 val INVERT: Int\r\n     val INCR\_WRAP: Int\r\n     val DECR\_WRAP: Int\r\n     val VENDOR: Int\r\n  
 val RENDERER: Int\r\n     val VERSION: Int\r\n     val NEAREST: Int\r\n     val LINEAR: Int\r\n     val  
 NEAREST\_MIPMAP\_NEAREST: Int\r\n     val LINEAR\_MIPMAP\_NEAREST: Int\r\n     val  
 NEAREST\_MIPMAP\_LINEAR: Int\r\n     val LINEAR\_MIPMAP\_LINEAR: Int\r\n     val  
 TEXTURE\_MAG\_FILTER: Int\r\n     val TEXTURE\_MIN\_FILTER: Int\r\n     val TEXTURE\_WRAP\_S:  
 Int\r\n     val TEXTURE\_WRAP\_T: Int\r\n     val TEXTURE\_2D: Int\r\n     val TEXTURE: Int\r\n     val  
 TEXTURE\_CUBE\_MAP: Int\r\n     val TEXTURE\_BINDING\_CUBE\_MAP: Int\r\n     val  
 TEXTURE\_CUBE\_MAP\_POSITIVE\_X: Int\r\n     val TEXTURE\_CUBE\_MAP\_NEGATIVE\_X: Int\r\n     val  
 TEXTURE\_CUBE\_MAP\_POSITIVE\_Y: Int\r\n     val TEXTURE\_CUBE\_MAP\_NEGATIVE\_Y: Int\r\n     val  
 TEXTURE\_CUBE\_MAP\_POSITIVE\_Z: Int\r\n     val TEXTURE\_CUBE\_MAP\_NEGATIVE\_Z: Int\r\n     val  
 MAX\_CUBE\_MAP\_TEXTURE\_SIZE: Int\r\n     val TEXTURE0: Int\r\n     val TEXTURE1: Int\r\n     val  
 TEXTURE2: Int\r\n     val TEXTURE3: Int\r\n     val TEXTURE4: Int\r\n     val TEXTURE5: Int\r\n     val  
 TEXTURE6: Int\r\n     val TEXTURE7: Int\r\n     val TEXTURE8: Int\r\n     val TEXTURE9: Int\r\n     val  
 TEXTURE10: Int\r\n     val TEXTURE11: Int\r\n     val TEXTURE12: Int\r\n     val TEXTURE13: Int\r\n  
 val TEXTURE14: Int\r\n     val TEXTURE15: Int\r\n     val TEXTURE16: Int\r\n     val TEXTURE17: Int\r\n  
    val TEXTURE18: Int\r\n     val TEXTURE19: Int\r\n     val TEXTURE20: Int\r\n     val TEXTURE21:  
 Int\r\n     val TEXTURE22: Int\r\n     val TEXTURE23: Int\r\n     val TEXTURE24: Int\r\n     val  
 TEXTURE25: Int\r\n     val TEXTURE26: Int\r\n     val TEXTURE27: Int\r\n     val TEXTURE28: Int\r\n  
 val TEXTURE29: Int\r\n     val TEXTURE30: Int\r\n     val TEXTURE31: Int\r\n     val ACTIVE\_TEXTURE:  
 Int\r\n     val REPEAT: Int\r\n     val CLAMP\_TO\_EDGE: Int\r\n     val MIRRORED\_REPEAT: Int\r\n  
 val FLOAT\_VEC2: Int\r\n     val FLOAT\_VEC3: Int\r\n     val FLOAT\_VEC4: Int\r\n     val INT\_VEC2:  
 Int\r\n     val INT\_VEC3: Int\r\n     val INT\_VEC4: Int\r\n     val BOOL: Int\r\n     val BOOL\_VEC2: Int\r\n  
    val BOOL\_VEC3: Int\r\n     val BOOL\_VEC4: Int\r\n     val FLOAT\_MAT2: Int\r\n     val FLOAT\_MAT3:  
 Int\r\n     val FLOAT\_MAT4: Int\r\n     val SAMPLER\_2D: Int\r\n     val SAMPLER\_CUBE: Int\r\n     val  
 VERTEX\_ATTRIB\_ARRAY\_ENABLED: Int\r\n     val VERTEX\_ATTRIB\_ARRAY\_SIZE: Int\r\n     val  
 VERTEX\_ATTRIB\_ARRAY\_STRIDE: Int\r\n     val VERTEX\_ATTRIB\_ARRAY\_TYPE: Int\r\n     val  
 VERTEX\_ATTRIB\_ARRAY\_NORMALIZED: Int\r\n     val VERTEX\_ATTRIB\_ARRAY\_POINTER: Int\r\n  
 val VERTEX\_ATTRIB\_ARRAY\_BUFFER\_BINDING: Int\r\n     val  
 IMPLEMENTATION\_COLOR\_READ\_TYPE: Int\r\n     val IMPLEMENTATION\_COLOR\_READ\_FORMAT:  
 Int\r\n     val COMPILE\_STATUS: Int\r\n     val LOW\_FLOAT: Int\r\n     val MEDIUM\_FLOAT: Int\r\n  
 val HIGH\_FLOAT: Int\r\n     val LOW\_INT: Int\r\n     val MEDIUM\_INT: Int\r\n     val HIGH\_INT: Int\r\n  
    val FRAMEBUFFER: Int\r\n     val RENDERBUFFER: Int\r\n     val RGBA4: Int\r\n     val RGB5\_A1:  
 Int\r\n     val RGB565: Int\r\n     val DEPTH\_COMPONENT16: Int\r\n     val STENCIL\_INDEX: Int\r\n  
 val STENCIL\_INDEX8: Int\r\n     val DEPTH\_STENCIL: Int\r\n     val RENDERBUFFER\_WIDTH: Int\r\n  
 val RENDERBUFFER\_HEIGHT: Int\r\n     val RENDERBUFFER\_INTERNAL\_FORMAT: Int\r\n     val  
 RENDERBUFFER\_RED\_SIZE: Int\r\n     val RENDERBUFFER\_GREEN\_SIZE: Int\r\n     val  
 RENDERBUFFER\_BLUE\_SIZE: Int\r\n     val RENDERBUFFER\_ALPHA\_SIZE: Int\r\n     val  
 RENDERBUFFER\_DEPTH\_SIZE: Int\r\n     val RENDERBUFFER\_STENCIL\_SIZE: Int\r\n     val  
 FRAMEBUFFER\_ATTACHMENT\_OBJECT\_TYPE: Int\r\n     val

```

FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int\r\n val
FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int\r\n val
FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int\r\n val COLOR_ATTACHMENT0:
Int\r\n val DEPTH_ATTACHMENT: Int\r\n val STENCIL_ATTACHMENT: Int\r\n val
DEPTH_STENCIL_ATTACHMENT: Int\r\n val NONE: Int\r\n val FRAMEBUFFER_COMPLETE:
Int\r\n val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int\r\n val
FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int\r\n val
FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int\r\n val FRAMEBUFFER_UNSUPPORTED: Int\r\n
val FRAMEBUFFER_BINDING: Int\r\n val RENDERBUFFER_BINDING: Int\r\n val
MAX_RENDERBUFFER_SIZE: Int\r\n val INVALID_FRAMEBUFFER_OPERATION: Int\r\n val
UNPACK_FLIP_Y_WEBGL: Int\r\n val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int\r\n val
CONTEXT_LOST_WEBGL: Int\r\n val UNPACK_COLORSPACE_CONVERSION_WEBGL: Int\r\n val
BROWSER_DEFAULT_WEBGL: Int\r\n }\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[WebGLContextEvent](https://developer.mozilla.org/en/docs/Web/API/WebGLContextEvent) to Kotlin\r\n
*/\r\n\r\npublic external open class WebGLContextEvent(type: String, eventInit: WebGLContextEventInit =
definedExternally) : Event {\r\n open val statusMessage: String\r\n}\r\n\r\n\r\npublic external interface
WebGLContextEventInit : EventInit {\r\n var statusMessage: String? /* = \"\" */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun
WebGLContextEventInit(statusMessage: String? = \"\", bubbles: Boolean? = false, cancelable: Boolean? = false,
composed: Boolean? = false): WebGLContextEventInit {\r\n val o = js(\"({})\")\r\n\r\n o[\"statusMessage\"] =
statusMessage\r\n o[\"bubbles\"] = bubbles\r\n o[\"cancelable\"] = cancelable\r\n o[\"composed\"] =
composed\r\n\r\n return o\r\n}\r\n\r\n\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[ArrayBuffer](https://developer.mozilla.org/en/docs/Web/API/ArrayBuffer) to Kotlin\r\n
*/\r\n\r\npublic external open
class ArrayBuffer(length: Int) : BufferDataSource {\r\n open val byteLength: Int\r\n fun slice(begin: Int, end: Int
= definedExternally): ArrayBuffer\r\n\r\n companion object {\r\n fun isView(value: Any?): Boolean\r\n
 }\r\n}\r\n
*/\r\n * Exposes the JavaScript
[ArrayBufferView](https://developer.mozilla.org/en/docs/Web/API/ArrayBufferView) to Kotlin\r\n
*/\r\n\r\npublic
external interface ArrayBufferView : BufferDataSource {\r\n val buffer: ArrayBuffer\r\n val byteOffset: Int\r\n
val byteLength: Int\r\n}\r\n
*/\r\n * Exposes the JavaScript
[Int8Array](https://developer.mozilla.org/en/docs/Web/API/Int8Array) to Kotlin\r\n
*/\r\n\r\npublic external open class
Int8Array : ArrayBufferView {\r\n constructor(length: Int)\r\n constructor(array: Int8Array)\r\n
 constructor(array: Array<Byte>)\r\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length:
Int = definedExternally)\r\n open val length: Int\r\n override val buffer: ArrayBuffer\r\n override val
byteOffset: Int\r\n override val byteLength: Int\r\n fun set(array: Int8Array, offset: Int = definedExternally):
Unit\r\n\r\n fun set(array: Array<Byte>, offset: Int = definedExternally): Unit\r\n fun subarray(start: Int, end: Int):
Int8Array\r\n\r\n companion object {\r\n val BYTES_PER_ELEMENT: Int\r\n
 }\r\n}\r\n
*/\r\n * Exposes the JavaScript
[Uint8Array](https://developer.mozilla.org/en/docs/Web/API/Uint8Array) to Kotlin\r\n
*/\r\n\r\npublic external open
class Uint8Array : ArrayBufferView {\r\n constructor(length: Int)\r\n constructor(array: Uint8Array)\r\n
 constructor(array: Array<Byte>)\r\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length:
Int = definedExternally)\r\n open val length: Int\r\n override val buffer: ArrayBuffer\r\n override val
byteOffset: Int\r\n override val byteLength: Int\r\n fun set(array: Uint8Array, offset: Int = definedExternally):
Unit\r\n\r\n fun set(array: Array<Byte>, offset: Int = definedExternally): Unit\r\n fun subarray(start: Int, end: Int):
Uint8Array\r\n\r\n companion object {\r\n val BYTES_PER_ELEMENT: Int\r\n
 }\r\n}\r\n
*/\r\n * Exposes the JavaScript

```

```

{ asDynamic()[index] = value; }
}

* Exposes the JavaScript
[UInt8ClampedArray](https://developer.mozilla.org/en/docs/Web/API/UInt8ClampedArray) to Kotlin
*

public external open class UInt8ClampedArray : ArrayBufferView {
 constructor(length: Int)
 constructor(array: UInt8ClampedArray)
 constructor(array: Array<Byte>)
 constructor(buffer:
ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
 open val length: Int
 override val buffer: ArrayBuffer
 override val byteOffset: Int
 override val byteLength: Int
 fun
set(array: UInt8ClampedArray, offset: Int = definedExternally): Unit
 fun set(array: Array<Byte>, offset: Int =
definedExternally): Unit
 fun subarray(start: Int, end: Int): UInt8ClampedArray

 companion object {
 val BYTES_PER_ELEMENT: Int
 }
}

@kotlin.internal.InlineOnly inline operator fun
UInt8ClampedArray.get(index: Int): Byte = asDynamic()[index]
}

@kotlin.internal.InlineOnly inline operator fun
UInt8ClampedArray.set(index: Int, value: Byte): Unit {
 asDynamic()[index] = value;
}

* Exposes the
JavaScript [Int16Array](https://developer.mozilla.org/en/docs/Web/API/Int16Array) to Kotlin
*

public external open class Int16Array : ArrayBufferView {
 constructor(length: Int)
 constructor(array:
Int16Array)
 constructor(array: Array<Short>)
 constructor(buffer: ArrayBuffer, byteOffset: Int =
definedExternally, length: Int = definedExternally)
 open val length: Int
 override val buffer:
ArrayBuffer
 override val byteOffset: Int
 override val byteLength: Int
 fun set(array: Int16Array,
offset: Int = definedExternally): Unit
 fun set(array: Array<Short>, offset: Int = definedExternally): Unit
 fun subarray(start: Int, end: Int): Int16Array

 companion object {
 val BYTES_PER_ELEMENT:
Int
 }
}

@kotlin.internal.InlineOnly inline operator fun Int16Array.get(index: Int): Short =
asDynamic()[index]
}

@kotlin.internal.InlineOnly inline operator fun Int16Array.set(index: Int, value: Short): Unit
{
 asDynamic()[index] = value;
}

* Exposes the JavaScript
[UInt16Array](https://developer.mozilla.org/en/docs/Web/API/UInt16Array) to Kotlin
*

public external open class UInt16Array : ArrayBufferView {
 constructor(length: Int)
 constructor(array: UInt16Array)
 constructor(array: Array<Short>)
 constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length:
Int = definedExternally)
 open val length: Int
 override val buffer: ArrayBuffer
 override val
byteOffset: Int
 override val byteLength: Int
 fun set(array: UInt16Array, offset: Int = definedExternally):
Unit
 fun set(array: Array<Short>, offset: Int = definedExternally): Unit
 fun subarray(start: Int, end: Int):
UInt16Array

 companion object {
 val BYTES_PER_ELEMENT: Int
 }
}

@kotlin.internal.InlineOnly inline operator fun UInt16Array.get(index: Int): Short =
asDynamic()[index]
}

@kotlin.internal.InlineOnly inline operator fun UInt16Array.set(index: Int, value: Short):
Unit {
 asDynamic()[index] = value;
}

* Exposes the JavaScript
[Int32Array](https://developer.mozilla.org/en/docs/Web/API/Int32Array) to Kotlin
*

public external open class Int32Array : ArrayBufferView {
 constructor(length: Int)
 constructor(array: Int32Array)
 constructor(array: Array<Int>)
 constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int
= definedExternally)
 open val length: Int
 override val buffer: ArrayBuffer
 override val byteOffset:
Int
 override val byteLength: Int
 fun set(array: Int32Array, offset: Int = definedExternally): Unit
 fun set(array: Array<Int>, offset: Int = definedExternally): Unit
 fun subarray(start: Int, end: Int): Int32Array

 companion object {
 val BYTES_PER_ELEMENT: Int
 }
}

@kotlin.internal.InlineOnly inline operator fun Int32Array.get(index: Int): Int = asDynamic()[index]
}

@kotlin.internal.InlineOnly inline operator fun Int32Array.set(index: Int, value: Int): Unit {
 asDynamic()[index] = value;
}

* Exposes the JavaScript
[UInt32Array](https://developer.mozilla.org/en/docs/Web/API/UInt32Array) to Kotlin
*

public external open class UInt32Array : ArrayBufferView {
 constructor(length: Int)
 constructor(array: UInt32Array)
 constructor(array: Array<Int>)
 constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int
= definedExternally)
 open val length: Int
 override val buffer: ArrayBuffer
 override val byteOffset:
Int
 override val byteLength: Int
 fun set(array: UInt32Array, offset: Int = definedExternally): Unit
 fun set(array: Array<Int>, offset: Int = definedExternally): Unit
 fun subarray(start: Int, end: Int):
UInt32Array

 companion object {
 val BYTES_PER_ELEMENT: Int
 }
}

@kotlin.internal.InlineOnly inline operator fun UInt32Array.get(index: Int): Int =

```

```

asDynamic()[index]
@kotlin.internal.InlineOnly inline operator fun Uint32Array.set(index: Int, value: Int): Unit
{ asDynamic()[index] = value; }
Exposes the JavaScript
[Float32Array](https://developer.mozilla.org/en/docs/Web/API/Float32Array) to Kotlin
public external
open class Float32Array : ArrayBufferView {
 constructor(length: Int)
 constructor(array:
Float32Array)
 constructor(array: Array<Float>)
 constructor(buffer: ArrayBuffer, byteOffset: Int =
definedExternally, length: Int = definedExternally)
 open val length: Int
 override val buffer:
ArrayBuffer
 override val byteOffset: Int
 override val byteLength: Int
 fun set(array: Float32Array,
offset: Int = definedExternally): Unit
 fun set(array: Array<Float>, offset: Int = definedExternally): Unit
 fun subarray(start: Int, end: Int): Float32Array
 companion object {
 val BYTES_PER_ELEMENT:
Int
 }
}
@kotlin.internal.InlineOnly inline operator fun Float32Array.get(index: Int): Float =
asDynamic()[index]
@kotlin.internal.InlineOnly inline operator fun Float32Array.set(index: Int, value: Float):
Unit { asDynamic()[index] = value; }
Exposes the JavaScript
[Float64Array](https://developer.mozilla.org/en/docs/Web/API/Float64Array) to Kotlin
public external
open class Float64Array : ArrayBufferView {
 constructor(length: Int)
 constructor(array:
Float64Array)
 constructor(array: Array<Double>)
 constructor(buffer: ArrayBuffer, byteOffset: Int =
definedExternally, length: Int = definedExternally)
 open val length: Int
 override val buffer:
ArrayBuffer
 override val byteOffset: Int
 override val byteLength: Int
 fun set(array: Float64Array,
offset: Int = definedExternally): Unit
 fun set(array: Array<Double>, offset: Int = definedExternally): Unit
 fun subarray(start: Int, end: Int): Float64Array
 companion object {
 val BYTES_PER_ELEMENT:
Int
 }
}
@kotlin.internal.InlineOnly inline operator fun Float64Array.get(index: Int): Double =
asDynamic()[index]
@kotlin.internal.InlineOnly inline operator fun Float64Array.set(index: Int, value: Double):
Unit { asDynamic()[index] = value; }
Exposes the JavaScript
[DataView](https://developer.mozilla.org/en/docs/Web/API/DataView) to Kotlin
public external open class
DataView(buffer: ArrayBuffer, byteOffset: Int = definedExternally, byteLength: Int = definedExternally) :
ArrayBufferView {
 override val buffer: ArrayBuffer
 override val byteOffset: Int
 override val
byteLength: Int
 fun getInt8(byteOffset: Int): Byte
 fun getUint8(byteOffset: Int): Byte
 fun
getInt16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
 fun getUint16(byteOffset: Int,
littleEndian: Boolean = definedExternally): Short
 fun getInt32(byteOffset: Int, littleEndian: Boolean =
definedExternally): Int
 fun getUint32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
 fun
getFloat32(byteOffset: Int, littleEndian: Boolean = definedExternally): Float
 fun getFloat64(byteOffset: Int,
littleEndian: Boolean = definedExternally): Double
 fun setInt8(byteOffset: Int, value: Byte): Unit
 fun
setUint8(byteOffset: Int, value: Byte): Unit
 fun setInt16(byteOffset: Int, value: Short, littleEndian: Boolean =
definedExternally): Unit
 fun setUint16(byteOffset: Int, value: Short, littleEndian: Boolean =
definedExternally): Unit
 fun setInt32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally):
Unit
 fun setUint32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally): Unit
 fun
setFloat32(byteOffset: Int, value: Float, littleEndian: Boolean = definedExternally): Unit
 fun
setFloat64(byteOffset: Int, value: Double, littleEndian: Boolean = definedExternally): Unit
}
public
external @marker interface BufferDataSource
public external @marker interface TexImageSource
}

/* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See libraries/tools/idl2k for
details
@file:Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
package
org.w3c.dom.clipboard
import kotlin.js.*
import org.khronos.webgl.*
import
org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.css.*
import
org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import
org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import
org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import
org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*
public external interface

```

```

ClipboardEventInit : EventInit {\r\n var clipboardData: DataTransfer? /* = null */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun
ClipboardEventInit(clipboardData: DataTransfer? = null, bubbles: Boolean? = false, cancelable: Boolean? = false,
composed: Boolean? = false): ClipboardEventInit {\r\n val o = js("{}")\r\n\r\n o["clipboardData"] =
clipboardData\r\n o["bubbles"] = bubbles\r\n o["cancelable"] = cancelable\r\n o["composed"] =
composed\r\n\r\n return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[ClipboardEvent](https://developer.mozilla.org/en/docs/Web/API/ClipboardEvent) to Kotlin\r\n */\r\n\r\npublic
external open class ClipboardEvent(type: String, eventInitDict: ClipboardEventInit = definedExternally) : Event
{\r\n open val clipboardData: DataTransfer?\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[Clipboard](https://developer.mozilla.org/en/docs/Web/API/Clipboard) to Kotlin\r\n */\r\n\r\npublic external abstract
class Clipboard : EventTarget {\r\n fun read(): Promise<DataTransfer>\r\n fun readText(): Promise<String>\r\n
fun write(data: DataTransfer): Promise<Unit>\r\n fun writeText(data: String): Promise<Unit>\r\n}\r\n\r\n\r\npublic
external interface ClipboardPermissionDescriptor {\r\n var allowWithoutGesture: Boolean? /* = false */\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic
inline fun ClipboardPermissionDescriptor(allowWithoutGesture: Boolean? = false): ClipboardPermissionDescriptor
{\r\n val o = js("{}")\r\n\r\n o["allowWithoutGesture"] = allowWithoutGesture\r\n\r\n return
o\r\n}\r\n\r\n\r\n", /*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n *
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*/\r\n\r\n\r\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\r\n\r\n// See libraries/tools/idl2k for
details\r\n\r\n\r\n@file:Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\r\n\r\npackage
org.w3c.dom.css\r\n\r\nimport kotlin.js.*\r\nimport org.khronos.webgl.*\r\nimport org.w3c.css.masking.*\r\nimport
org.w3c.dom.*\r\nimport org.w3c.dom.clipboard.*\r\nimport org.w3c.dom.events.*\r\nimport
org.w3c.dom.mediacapture.*\r\nimport org.w3c.dom.parsing.*\r\nimport org.w3c.dom.pointerevents.*\r\nimport
org.w3c.dom.svg.*\r\nimport org.w3c.dom.url.*\r\nimport org.w3c.fetch.*\r\nimport org.w3c.files.*\r\nimport
org.w3c.notifications.*\r\nimport org.w3c.performance.*\r\nimport org.w3c.workers.*\r\nimport
org.w3c.xhr.*\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[CSSStyleDeclaration](https://developer.mozilla.org/en/docs/Web/API/CSSStyleDeclaration) to Kotlin\r\n
*/\r\n\r\n\r\npublic external abstract class CSSStyleDeclaration : ItemArrayLike<String> {\r\n open var cssText:
String\r\n override val length: Int\r\n open val parentRule: CSSRule?\r\n open var cssFloat: String\r\n open
var _dashed_attribute: String\r\n open var _camel_cased_attribute: String\r\n open var _webkit_cased_attribute:
String\r\n open var alignContent: String\r\n open var alignItems: String\r\n open var alignSelf: String\r\n
open var animation: String\r\n open var animationDelay: String\r\n open var animationDirection: String\r\n
open var animationDuration: String\r\n open var animationFillMode: String\r\n open var
animationIterationCount: String\r\n open var animationName: String\r\n open var animationPlayState: String\r\n
open var animationTimingFunction: String\r\n open var backfaceVisibility: String\r\n open var background:
String\r\n open var backgroundAttachment: String\r\n open var backgroundClip: String\r\n open var
backgroundColor: String\r\n open var backgroundImage: String\r\n open var backgroundOrigin: String\r\n
open var backgroundPosition: String\r\n open var backgroundRepeat: String\r\n open var backgroundSize:
String\r\n open var border: String\r\n open var borderBottom: String\r\n open var borderBottomColor:
String\r\n open var borderBottomLeftRadius: String\r\n open var borderBottomRightRadius: String\r\n open
var borderBottomStyle: String\r\n open var borderBottomWidth: String\r\n open var borderCollapse: String\r\n
open var borderColor: String\r\n open var borderImage: String\r\n open var borderImageOutset: String\r\n
open var borderImageRepeat: String\r\n open var borderImageSlice: String\r\n open var borderImageSource:
String\r\n open var borderImageWidth: String\r\n open var borderLeft: String\r\n open var borderLeftColor:
String\r\n open var borderLeftStyle: String\r\n open var borderLeftWidth: String\r\n open var borderRadius:
String\r\n open var borderRight: String\r\n open var borderRightColor: String\r\n open var borderRightStyle:
String\r\n open var borderRightWidth: String\r\n open var borderSpacing: String\r\n open var borderStyle:
String\r\n open var borderTop: String\r\n open var borderTopColor: String\r\n open var borderTopLeftRadius:

```

String\r\n open var borderTopRightRadius: String\r\n open var borderTopStyle: String\r\n open var  
borderTopWidth: String\r\n open var borderWidth: String\r\n open var bottom: String\r\n open var  
boxDecorationBreak: String\r\n open var boxShadow: String\r\n open var boxSizing: String\r\n open var  
breakAfter: String\r\n open var breakBefore: String\r\n open var breakInside: String\r\n open var captionSide:  
String\r\n open var clear: String\r\n open var clip: String\r\n open var color: String\r\n open var  
columnCount: String\r\n open var columnFill: String\r\n open var columnGap: String\r\n open var  
columnRule: String\r\n open var columnRuleColor: String\r\n open var columnRuleStyle: String\r\n open var  
columnRuleWidth: String\r\n open var columnSpan: String\r\n open var columnWidth: String\r\n open var  
columns: String\r\n open var content: String\r\n open var counterIncrement: String\r\n open var counterReset:  
String\r\n open var cursor: String\r\n open var direction: String\r\n open var display: String\r\n open var  
emptyCells: String\r\n open var filter: String\r\n open var flex: String\r\n open var flexBasis: String\r\n open  
var flexDirection: String\r\n open var flexFlow: String\r\n open var flexGrow: String\r\n open var flexShrink:  
String\r\n open var flexWrap: String\r\n open var font: String\r\n open var fontFamily: String\r\n open var  
fontFeatureSettings: String\r\n open var fontKerning: String\r\n open var fontLanguageOverride: String\r\n  
open var fontSize: String\r\n open var fontSizeAdjust: String\r\n open var fontStretch: String\r\n open var  
fontStyle: String\r\n open var fontSynthesis: String\r\n open var fontVariant: String\r\n open var  
fontVariantAlternates: String\r\n open var fontVariantCaps: String\r\n open var fontVariantEastAsian: String\r\n  
open var fontVariantLigatures: String\r\n open var fontVariantNumeric: String\r\n open var fontVariantPosition:  
String\r\n open var fontWeight: String\r\n open var hangingPunctuation: String\r\n open var height: String\r\n  
open var hyphens: String\r\n open var imageOrientation: String\r\n open var imageRendering: String\r\n open  
var imageResolution: String\r\n open var imeMode: String\r\n open var justifyContent: String\r\n open var left:  
String\r\n open var letterSpacing: String\r\n open var lineBreak: String\r\n open var lineHeight: String\r\n  
open var listStyle: String\r\n open var listStyleImage: String\r\n open var listStylePosition: String\r\n open var  
listStyleType: String\r\n open var margin: String\r\n open var marginBottom: String\r\n open var marginLeft:  
String\r\n open var marginRight: String\r\n open var marginTop: String\r\n open var mark: String\r\n open  
var markAfter: String\r\n open var markBefore: String\r\n open var marks: String\r\n open var  
marqueeDirection: String\r\n open var marqueePlayCount: String\r\n open var marqueeSpeed: String\r\n open  
var marqueeStyle: String\r\n open var mask: String\r\n open var maskType: String\r\n open var maxHeight:  
String\r\n open var maxWidth: String\r\n open var minHeight: String\r\n open var minWidth: String\r\n open  
var navDown: String\r\n open var navIndex: String\r\n open var navLeft: String\r\n open var navRight:  
String\r\n open var navUp: String\r\n open var objectFit: String\r\n open var objectPosition: String\r\n open  
var opacity: String\r\n open var order: String\r\n open var orphans: String\r\n open var outline: String\r\n  
open var outlineColor: String\r\n open var outlineOffset: String\r\n open var outlineStyle: String\r\n open var  
outlineWidth: String\r\n open var overflowWrap: String\r\n open var overflowX: String\r\n open var  
overflowY: String\r\n open var padding: String\r\n open var paddingBottom: String\r\n open var paddingLeft:  
String\r\n open var paddingRight: String\r\n open var paddingTop: String\r\n open var pageBreakAfter:  
String\r\n open var pageBreakBefore: String\r\n open var pageBreakInside: String\r\n open var perspective:  
String\r\n open var perspectiveOrigin: String\r\n open var phonemes: String\r\n open var position: String\r\n  
open var quotes: String\r\n open var resize: String\r\n open var rest: String\r\n open var restAfter: String\r\n  
open var restBefore: String\r\n open var right: String\r\n open var tabSize: String\r\n open var tableLayout:  
String\r\n open var textAlign: String\r\n open var textAlignLast: String\r\n open var textCombineUpright:  
String\r\n open var textDecoration: String\r\n open var textDecorationColor: String\r\n open var  
textDecorationLine: String\r\n open var textDecorationStyle: String\r\n open var textIndent: String\r\n open  
var textJustify: String\r\n open var textOrientation: String\r\n open var textOverflow: String\r\n open var  
textShadow: String\r\n open var textTransform: String\r\n open var textUnderlinePosition: String\r\n open var  
top: String\r\n open var transform: String\r\n open var transformOrigin: String\r\n open var transformStyle:  
String\r\n open var transition: String\r\n open var transitionDelay: String\r\n open var transitionDuration:  
String\r\n open var transitionProperty: String\r\n open var transitionTimingFunction: String\r\n open var

```

unicodeBidi: String\r\n open var verticalAlign: String\r\n open var visibility: String\r\n open var voiceBalance:
String\r\n open var voiceDuration: String\r\n open var voicePitch: String\r\n open var voicePitchRange:
String\r\n open var voiceRate: String\r\n open var voiceStress: String\r\n open var voiceVolume: String\r\n
open var whiteSpace: String\r\n open var widows: String\r\n open var width: String\r\n open var wordBreak:
String\r\n open var wordSpacing: String\r\n open var wordWrap: String\r\n open var writingMode: String\r\n
open var zIndex: String\r\n override fun item(index: Int): String\r\n fun getPropertyValue(property: String):
String\r\n fun getPropertyPriority(property: String): String\r\n fun setProperty(property: String, value: String,
priority: String = definedExternally): Unit\r\n fun setPropertyValue(property: String, value: String): Unit\r\n fun
setPropertyPriority(property: String, priority: String): Unit\r\n fun removeProperty(property: String):
String\r\n}\r\n@kotlin.internal.InlineOnly inline operator fun CSSStyleDeclaration.get(index: Int): String? =
asDynamic()[index]\r\n\r\npublic external abstract class MediaList : ItemArrayLike<String> {\r\n open var
mediaText: String\r\n override val length: Int\r\n override fun item(index: Int): String?\r\n fun
appendMedium(medium: String): Unit\r\n fun deleteMedium(medium: String):
Unit\r\n}\r\n@kotlin.internal.InlineOnly inline operator fun MediaList.get(index: Int): String? =
asDynamic()[index]\r\n\r\n**\r\n * Exposes the JavaScript
[StyleSheet](https://developer.mozilla.org/en/docs/Web/API/StyleSheet) to Kotlin\r\n */\r\npublic external abstract
class StyleSheet {\r\n open val type: String\r\n open val href: String?\r\n open val ownerNode:
UnionElementOrProcessingInstruction?\r\n open val parentStyleSheet: StyleSheet?\r\n open val title: String?\r\n
open val media: MediaList\r\n open var disabled: Boolean\r\n}\r\n\r\n**\r\n * Exposes the JavaScript
[CSSStyleSheet](https://developer.mozilla.org/en/docs/Web/API/CSSStyleSheet) to Kotlin\r\n */\r\npublic external
abstract class CSSStyleSheet : StyleSheet {\r\n open val ownerRule: CSSRule?\r\n open val cssRules:
CSSRuleList\r\n fun insertRule(rule: String, index: Int): Int\r\n fun deleteRule(index: Int):
Unit\r\n}\r\n\r\n**\r\n * Exposes the JavaScript
[StyleSheetList](https://developer.mozilla.org/en/docs/Web/API/StyleSheetList) to Kotlin\r\n */\r\npublic external
abstract class StyleSheetList : ItemArrayLike<StyleSheet> {\r\n override val length: Int\r\n override fun
item(index: Int): StyleSheet?\r\n}\r\n@kotlin.internal.InlineOnly inline operator fun StyleSheetList.get(index: Int):
StyleSheet? = asDynamic()[index]\r\n\r\n**\r\n * Exposes the JavaScript
[LinkStyle](https://developer.mozilla.org/en/docs/Web/API/LinkStyle) to Kotlin\r\n */\r\npublic external interface
LinkStyle {\r\n val sheet: StyleSheet?\r\n}\r\n\r\n**\r\n * Exposes the JavaScript
[CSSRuleList](https://developer.mozilla.org/en/docs/Web/API/CSSRuleList) to Kotlin\r\n */\r\npublic external
abstract class CSSRuleList : ItemArrayLike<CSSRule> {\r\n override val length: Int\r\n override fun
item(index: Int): CSSRule?\r\n}\r\n@kotlin.internal.InlineOnly inline operator fun CSSRuleList.get(index: Int):
CSSRule? = asDynamic()[index]\r\n\r\n**\r\n * Exposes the JavaScript
[CSSRule](https://developer.mozilla.org/en/docs/Web/API/CSSRule) to Kotlin\r\n */\r\npublic external abstract
class CSSRule {\r\n open val type: Short\r\n open var cssText: String\r\n open val parentRule: CSSRule?\r\n
open val parentStyleSheet: CSSStyleSheet?\r\n\r\n companion object {\r\n val STYLE_RULE: Short\r\n
val CHARSET_RULE: Short\r\n val IMPORT_RULE: Short\r\n val MEDIA_RULE: Short\r\n val
FONT_FACE_RULE: Short\r\n val PAGE_RULE: Short\r\n val MARGIN_RULE: Short\r\n val
NAMESPACE_RULE: Short\r\n }\r\n}\r\n\r\n**\r\n * Exposes the JavaScript
[CSSStyleRule](https://developer.mozilla.org/en/docs/Web/API/CSSStyleRule) to Kotlin\r\n */\r\npublic external
abstract class CSSStyleRule : CSSRule {\r\n open var selectorText: String\r\n open val style:
CSSStyleDeclaration\r\n}\r\n\r\npublic external abstract class CSSImportRule : CSSRule {\r\n open val href:
String\r\n open val media: MediaList\r\n open val styleSheet: CSSStyleSheet\r\n}\r\n\r\n**\r\n * Exposes the
JavaScript [CSSGroupingRule](https://developer.mozilla.org/en/docs/Web/API/CSSGroupingRule) to Kotlin\r\n
*/\r\npublic external abstract class CSSGroupingRule : CSSRule {\r\n open val cssRules: CSSRuleList\r\n fun
insertRule(rule: String, index: Int): Int\r\n fun deleteRule(index: Int): Unit\r\n}\r\n\r\n**\r\n * Exposes the
JavaScript [CSSMediaRule](https://developer.mozilla.org/en/docs/Web/API/CSSMediaRule) to Kotlin\r\n
*/\r\npublic external abstract class CSSMediaRule : CSSGroupingRule {\r\n open val media:

```





```

get() = definedExternally\r\n set(value) = definedExternally\r\n var clientX: Int? /* = 0 */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var clientY: Int? /* = 0 */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var button: Short? /* = 0 */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var buttons: Short? /* = 0 */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var relatedTarget: EventTarget? /* = null */\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic
inline fun MouseEventInit(screenX: Int? = 0, screenY: Int? = 0, clientX: Int? = 0, clientY: Int? = 0, button: Short? =
0, buttons: Short? = 0, relatedTarget: EventTarget? = null, ctrlKey: Boolean? = false, shiftKey: Boolean? = false,
altKey: Boolean? = false, metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock:
Boolean? = false, modifierFn: Boolean? = false, modifierFnLock: Boolean? = false, modifierHyper: Boolean? =
false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false,
modifierSymbol: Boolean? = false, modifierSymbolLock: Boolean? = false, view: Window? = null, detail: Int? = 0,
bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): MouseEventInit {\r\n val o
= js("{}")\r\n\r\n o["screenX"] = screenX\r\n o["screenY"] = screenY\r\n o["clientX"] = clientX\r\n
o["clientY"] = clientY\r\n o["button"] = button\r\n o["buttons"] = buttons\r\n o["relatedTarget"] =
relatedTarget\r\n o["ctrlKey"] = ctrlKey\r\n o["shiftKey"] = shiftKey\r\n o["altKey"] = altKey\r\n
o["metaKey"] = metaKey\r\n o["modifierAltGraph"] = modifierAltGraph\r\n o["modifierCapsLock"] =
modifierCapsLock\r\n o["modifierFn"] = modifierFn\r\n o["modifierFnLock"] = modifierFnLock\r\n
o["modifierHyper"] = modifierHyper\r\n o["modifierNumLock"] = modifierNumLock\r\n
o["modifierScrollLock"] = modifierScrollLock\r\n o["modifierSuper"] = modifierSuper\r\n
o["modifierSymbol"] = modifierSymbol\r\n o["modifierSymbolLock"] = modifierSymbolLock\r\n o["view"]
= view\r\n o["detail"] = detail\r\n o["bubbles"] = bubbles\r\n o["cancelable"] = cancelable\r\n
o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\npublic external interface EventModifierInit : UIEventInit
{\r\n var ctrlKey: Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var shiftKey: Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var altKey: Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var metaKey: Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var modifierAltGraph: Boolean? /* = false */\r\n get() = definedExternally\r\n
set(value) = definedExternally\r\n var modifierCapsLock: Boolean? /* = false */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var modifierFn: Boolean? /* = false */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var modifierFnLock: Boolean? /* = false */\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n var modifierHyper: Boolean? /* = false */\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n var modifierNumLock: Boolean? /* = false
*/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var modifierScrollLock: Boolean?
/* = false */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var modifierSuper:
Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var
modifierSymbol: Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n
var modifierSymbolLock: Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun EventModifierInit(ctrlKey: Boolean?
= false, shiftKey: Boolean? = false, altKey: Boolean? = false, metaKey: Boolean? = false, modifierAltGraph:
Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn: Boolean? = false, modifierFnLock: Boolean? =
false, modifierHyper: Boolean? = false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false,
modifierSuper: Boolean? = false, modifierSymbol: Boolean? = false, modifierSymbolLock: Boolean? = false, view:
Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? =
false): EventModifierInit {\r\n val o = js("{}")\r\n\r\n o["ctrlKey"] = ctrlKey\r\n o["shiftKey"] =
shiftKey\r\n o["altKey"] = altKey\r\n o["metaKey"] = metaKey\r\n o["modifierAltGraph"] =
modifierAltGraph\r\n o["modifierCapsLock"] = modifierCapsLock\r\n o["modifierFn"] = modifierFn\r\n
o["modifierFnLock"] = modifierFnLock\r\n o["modifierHyper"] = modifierHyper\r\n

```

```

o["modifierNumLock"] = modifierNumLock\r\n o["modifierScrollLock"] = modifierScrollLock\r\n
o["modifierSuper"] = modifierSuper\r\n o["modifierSymbol"] = modifierSymbol\r\n
o["modifierSymbolLock"] = modifierSymbolLock\r\n o["view"] = view\r\n o["detail"] = detail\r\n
o["bubbles"] = bubbles\r\n o["cancelable"] = cancelable\r\n o["composed"] = composed\r\n\r\n return
o\r\n}\r\n\r\n\r\n**\r\n * Exposes the JavaScript
[WheelEvent](https://developer.mozilla.org/en/docs/Web/API/WheelEvent) to Kotlin\r\n *^\r\n\r\npublic external open
class WheelEvent(type: String, eventInitDict: WheelEventInit = definedExternally) : MouseEvent {\r\n open val
deltaX: Double\r\n open val deltaY: Double\r\n open val deltaZ: Double\r\n open val deltaMode: Int\r\n\r\n companion object {\r\n val DOM_DELTA_PIXEL: Int\r\n val DOM_DELTA_LINE: Int\r\n val
DOM_DELTA_PAGE: Int\r\n }\r\n}\r\n\r\n\r\npublic external interface WheelEventInit : MouseEventInit {\r\n var
deltaX: Double? /* = 0.0 */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var
deltaY: Double? /* = 0.0 */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var deltaZ:
Double? /* = 0.0 */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var deltaMode:
Int? /* = 0 */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun WheelEventInit(deltaX: Double? =
0.0, deltaY: Double? = 0.0, deltaZ: Double? = 0.0, deltaMode: Int? = 0, screenX: Int? = 0, screenY: Int? = 0,
clientX: Int? = 0, clientY: Int? = 0, button: Short? = 0, buttons: Short? = 0, relatedTarget: EventTarget? = null,
ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false, metaKey: Boolean? = false,
modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn: Boolean? = false,
modifierFnLock: Boolean? = false, modifierHyper: Boolean? = false, modifierNumLock: Boolean? = false,
modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false, modifierSymbol: Boolean? = false,
modifierSymbolLock: Boolean? = false, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false,
cancelable: Boolean? = false, composed: Boolean? = false): WheelEventInit {\r\n val o = js("{}")\r\n\r\n o["deltaX"] =
deltaX\r\n o["deltaY"] = deltaY\r\n o["deltaZ"] = deltaZ\r\n o["deltaMode"] =
deltaMode\r\n o["screenX"] = screenX\r\n o["screenY"] = screenY\r\n o["clientX"] = clientX\r\n
o["clientY"] = clientY\r\n o["button"] = button\r\n o["buttons"] = buttons\r\n o["relatedTarget"] =
relatedTarget\r\n o["ctrlKey"] = ctrlKey\r\n o["shiftKey"] = shiftKey\r\n o["altKey"] = altKey\r\n
o["metaKey"] = metaKey\r\n o["modifierAltGraph"] = modifierAltGraph\r\n o["modifierCapsLock"] =
modifierCapsLock\r\n o["modifierFn"] = modifierFn\r\n o["modifierFnLock"] = modifierFnLock\r\n
o["modifierHyper"] = modifierHyper\r\n o["modifierNumLock"] = modifierNumLock\r\n
o["modifierScrollLock"] = modifierScrollLock\r\n o["modifierSuper"] = modifierSuper\r\n
o["modifierSymbol"] = modifierSymbol\r\n o["modifierSymbolLock"] = modifierSymbolLock\r\n o["view"]
= view\r\n o["detail"] = detail\r\n o["bubbles"] = bubbles\r\n o["cancelable"] = cancelable\r\n
o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\n\r\n**\r\n * Exposes the JavaScript
[InputEvent](https://developer.mozilla.org/en/docs/Web/API/InputEvent) to Kotlin\r\n *^\r\n\r\npublic external open
class InputEvent(type: String, eventInitDict: InputEventInit = definedExternally) : UIEvent {\r\n open val data:
String\r\n open val isComposing: Boolean\r\n}\r\n\r\n\r\npublic external interface InputEventInit : UIEventInit {\r\n
var data: String? /* = "" */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var
isComposing: Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun InputEventInit(data: String? = "",
isComposing: Boolean? = false, view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable:
Boolean? = false, composed: Boolean? = false): InputEventInit {\r\n val o = js("{}")\r\n\r\n o["data"] =
data\r\n o["isComposing"] = isComposing\r\n o["view"] = view\r\n o["detail"] = detail\r\n o["bubbles"]
= bubbles\r\n o["cancelable"] = cancelable\r\n o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\n\r\n**\r\n * Exposes the JavaScript
[KeyboardEvent](https://developer.mozilla.org/en/docs/Web/API/KeyboardEvent) to
Kotlin\r\n *^\r\n\r\npublic external open class KeyboardEvent(type: String, eventInitDict: KeyboardEventInit =
definedExternally) : UIEvent {\r\n open val key: String\r\n open val code: String\r\n open val location: Int\r\n
open val ctrlKey: Boolean\r\n open val shiftKey: Boolean\r\n open val altKey: Boolean\r\n open val metaKey:

```

```
Boolean\r\n open val repeat: Boolean\r\n open val isComposing: Boolean\r\n open val charCode: Int\r\n open
val keyCode: Int\r\n open val which: Int\r\n fun getModifierState(keyArg: String): Boolean\r\n\r\n companion
object {\r\n val DOM_KEY_LOCATION_STANDARD: Int\r\n val DOM_KEY_LOCATION_LEFT:
Int\r\n val DOM_KEY_LOCATION_RIGHT: Int\r\n val DOM_KEY_LOCATION_NUMPAD: Int\r\n
}\r\n\r\n\r\npublic external interface KeyboardEventInit : EventModifierInit {\r\n val key: String? /* = \"\" */\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n var code: String? /* = \"\" */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var location: Int? /* = 0 */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var repeat: Boolean? /* = false */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var isComposing: Boolean? /* = false */\r\n get()
= definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline
fun KeyboardEventInit(key: String? = \"\", code: String? = \"\", location: Int? = 0, repeat: Boolean? = false,
isComposing: Boolean? = false, ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false,
metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn:
Boolean? = false, modifierFnLock: Boolean? = false, modifierHyper: Boolean? = false, modifierNumLock:
Boolean? = false, modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false, modifierSymbol:
Boolean? = false, modifierSymbolLock: Boolean? = false, view: Window? = null, detail: Int? = 0, bubbles:
Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): KeyboardEventInit {\r\n val o =
js(\"({})\")\r\n o[\"key\"] = key\r\n o[\"code\"] = code\r\n o[\"location\"] = location\r\n o[\"repeat\"] =
repeat\r\n o[\"isComposing\"] = isComposing\r\n o[\"ctrlKey\"] = ctrlKey\r\n o[\"shiftKey\"] = shiftKey\r\n
o[\"altKey\"] = altKey\r\n o[\"metaKey\"] = metaKey\r\n o[\"modifierAltGraph\"] = modifierAltGraph\r\n
o[\"modifierCapsLock\"] = modifierCapsLock\r\n o[\"modifierFn\"] = modifierFn\r\n o[\"modifierFnLock\"] =
modifierFnLock\r\n o[\"modifierHyper\"] = modifierHyper\r\n o[\"modifierNumLock\"] =
modifierNumLock\r\n o[\"modifierScrollLock\"] = modifierScrollLock\r\n o[\"modifierSuper\"] =
modifierSuper\r\n o[\"modifierSymbol\"] = modifierSymbol\r\n o[\"modifierSymbolLock\"] =
modifierSymbolLock\r\n o[\"view\"] = view\r\n o[\"detail\"] = detail\r\n o[\"bubbles\"] = bubbles\r\n
o[\"cancelable\"] = cancelable\r\n o[\"composed\"] = composed\r\n\r\n return o\r\n}\r\n\r\n/*\r\n * Exposes the
JavaScript [CompositionEvent](https://developer.mozilla.org/en/docs/Web/API/CompositionEvent) to Kotlin\r\n
*\r\n\r\npublic external open class CompositionEvent(type: String, eventInitDict: CompositionEventInit =
definedExternally) : UIEvent {\r\n open val data: String\r\n}\r\n\r\npublic external interface CompositionEventInit
: UIEventInit {\r\n var data: String? /* = \"\" */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun CompositionEventInit(data: String? =
\", view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false, cancelable: Boolean? = false, composed:
Boolean? = false): CompositionEventInit {\r\n val o = js(\"({})\")\r\n o[\"data\"] = data\r\n o[\"view\"] =
view\r\n o[\"detail\"] = detail\r\n o[\"bubbles\"] = bubbles\r\n o[\"cancelable\"] = cancelable\r\n
o[\"composed\"] = composed\r\n\r\n return o\r\n}\r\n\r\n/*\r\n * Exposes the JavaScript
[Event](https://developer.mozilla.org/en/docs/Web/API/Event) to Kotlin\r\n
*\r\n\r\npublic external open class
Event(type: String, eventInitDict: EventInit = definedExternally) {\r\n open val type: String\r\n open val target:
EventTarget?\r\n open val currentTarget: EventTarget?\r\n open val eventPhase: Short\r\n open val bubbles:
Boolean\r\n open val cancelable: Boolean\r\n open val defaultPrevented: Boolean\r\n open val composed:
Boolean\r\n open val isTrusted: Boolean\r\n open val timeStamp: Number\r\n fun composedPath():
Array<EventTarget>\r\n fun stopPropagation(): Unit\r\n fun stopImmediatePropagation(): Unit\r\n fun
preventDefault(): Unit\r\n fun initEvent(type: String, bubbles: Boolean, cancelable: Boolean): Unit\r\n\r\n
companion object {\r\n val NONE: Short\r\n val CAPTURING_PHASE: Short\r\n val AT_TARGET:
Short\r\n val BUBBLING_PHASE: Short\r\n }\r\n}\r\n\r\n/*\r\n * Exposes the JavaScript
[EventTarget](https://developer.mozilla.org/en/docs/Web/API/EventTarget) to Kotlin\r\n
*\r\n\r\npublic external
abstract class EventTarget {\r\n fun addEventListener(type: String, callback: EventListener?, options: dynamic =
definedExternally): Unit\r\n fun addEventListener(type: String, callback: ((Event) -> Unit)?, options: dynamic =
definedExternally): Unit\r\n fun removeEventListener(type: String, callback: EventListener?, options: dynamic =
```

definedExternally): Unit\r\n fun removeEventListener(type: String, callback: ((Event) -> Unit)?, options: dynamic = definedExternally): Unit\r\n fun dispatchEvent(event: Event): Boolean\r\n}\r\n\r\n/\*\*\r\n \* Exposes the JavaScript [EventListener](https://developer.mozilla.org/en/docs/Web/API/EventListener) to Kotlin\r\n \*/\r\npublic external interface EventListener {\r\n fun handleEvent(event: Event): Unit\r\n}\r\n\r\n","/\*\*\r\n \* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n \* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n \*/\r\n\r\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!\r\n\r\n// See libraries/tools/idl2k for details\r\n\r\n@file:Suppress("\r\nNESTED\_CLASS\_IN\_EXTERNAL\_INTERFACE\r\n")\r\n\r\npackage org.w3c.dom\r\n\r\nimport kotlin.js.\*\r\nimport org.khronos.webgl.\*\r\nimport org.w3c.css.masking.\*\r\nimport org.w3c.dom.clipboard.\*\r\nimport org.w3c.dom.css.\*\r\nimport org.w3c.dom.events.\*\r\nimport org.w3c.dom.mediacapture.\*\r\nimport org.w3c.dom.parsing.\*\r\nimport org.w3c.dom.pointerevents.\*\r\nimport org.w3c.dom.svg.\*\r\nimport org.w3c.dom.url.\*\r\nimport org.w3c.fetch.\*\r\nimport org.w3c.files.\*\r\nimport org.w3c.notifications.\*\r\nimport org.w3c.performance.\*\r\nimport org.w3c.workers.\*\r\nimport org.w3c.xhr.\*\r\n\r\n/\*\*\r\n \* Exposes the JavaScript [Navigator](https://developer.mozilla.org/en/docs/Web/API/Navigator) to Kotlin\r\n \*/\r\npublic external abstract class Navigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorContentUtils, NavigatorCookies, NavigatorPlugins, NavigatorConcurrentHardware {\r\n open val serviceWorker: ServiceWorkerContainer\r\n open val maxTouchPoints: Int\r\n open val mediaDevices: MediaDevices\r\n open val clipboard: Clipboard\r\n fun vibrate(pattern: dynamic): Boolean\r\n fun getUserMedia(constraints: MediaStreamConstraints, successCallback: (MediaStream) -> Unit, errorCallback: (dynamic) -> Unit): Unit\r\n}\r\n\r\n/\*\*\r\n \* Exposes the JavaScript [Document](https://developer.mozilla.org/en/docs/Web/API/Document) to Kotlin\r\n \*/\r\npublic external open class Document : Node, GlobalEventHandlers, DocumentAndElementEventHandlers, NonElementParentNode, DocumentOrShadowRoot, ParentNode, GeometryUtils {\r\n open val fullscreenEnabled: Boolean\r\n open val fullscreen: Boolean\r\n var onfullscreenchange: ((Event) -> dynamic)?\r\n var onfullscreenererror: ((Event) -> dynamic)?\r\n open val rootElement: SVGSVGElement?\r\n var title: String\r\n open val referrer: String\r\n var domain: String\r\n open val activeElement: Element?\r\n open val location: Location?\r\n var cookie: String\r\n open val lastModified: String\r\n open val readyState: DocumentReadyState\r\n var dir: String\r\n var body: HTMLElement?\r\n open val head: HTMLHeadElement?\r\n open val images: HTMLCollection\r\n open val embeds: HTMLCollection\r\n open val plugins: HTMLCollection\r\n open val links: HTMLCollection\r\n open val forms: HTMLCollection\r\n open val scripts: HTMLCollection\r\n open val currentScript: HTMLScriptElement?\r\n open val defaultView: Window?\r\n var designMode: String\r\n var onreadystatechange: ((Event) -> dynamic)?\r\n var fgColor: String\r\n var linkColor: String\r\n var vlinkColor: String\r\n var alinkColor: String\r\n var bgColor: String\r\n open val anchors: HTMLCollection\r\n open val applets: HTMLCollection\r\n open val all: HTMLAllCollection\r\n open val implementation: DOMImplementation\r\n open val URL: String\r\n open val documentURI: String\r\n open val origin: String\r\n open val compatMode: String\r\n open val characterSet: String\r\n open val charset: String\r\n open val inputEncoding: String\r\n open val contentType: String\r\n open val doctype: DocumentType?\r\n open val documentElement: Element?\r\n open val scrollingElement: Element?\r\n open val styleSheets: StyleSheetList\r\n override var ongotpointercapture: ((PointerEvent) -> dynamic)?\r\n override var onlostpointercapture: ((PointerEvent) -> dynamic)?\r\n override var onpointerdown: ((PointerEvent) -> dynamic)?\r\n override var onpointermove: ((PointerEvent) -> dynamic)?\r\n override var onpointerup: ((PointerEvent) -> dynamic)?\r\n override var onpointercancel: ((PointerEvent) -> dynamic)?\r\n override var onpointerover: ((PointerEvent) -> dynamic)?\r\n override var onpointerout: ((PointerEvent) -> dynamic)?\r\n override var onpointerenter: ((PointerEvent) -> dynamic)?\r\n override var onpointerleave: ((PointerEvent) -> dynamic)?\r\n override var onabort: ((Event) -> dynamic)?\r\n override var onblur: ((FocusEvent) -> dynamic)?\r\n override var oncancel: ((Event) -> dynamic)?\r\n override var oncanplay: ((Event) -> dynamic)?\r\n override var oncanplaythrough: ((Event) -> dynamic)?\r\n override var onchange: ((Event) -> dynamic)?\r\n override var onclick: ((MouseEvent) -> dynamic)?\r\n override var onclose: ((Event) -

```

> dynamic)?\r\n override var oncontextmenu: ((MouseEvent) -> dynamic)?\r\n override var oncuechange:
((Event) -> dynamic)?\r\n override var ondblclick: ((MouseEvent) -> dynamic)?\r\n override var ondrag:
((DragEvent) -> dynamic)?\r\n override var ondragend: ((DragEvent) -> dynamic)?\r\n override var ondragenter:
((DragEvent) -> dynamic)?\r\n override var ondragexit: ((DragEvent) -> dynamic)?\r\n override var
ondragleave: ((DragEvent) -> dynamic)?\r\n override var ondragover: ((DragEvent) -> dynamic)?\r\n override
var ondragstart: ((DragEvent) -> dynamic)?\r\n override var ondrop: ((DragEvent) -> dynamic)?\r\n override var
ondurationchange: ((Event) -> dynamic)?\r\n override var onemptied: ((Event) -> dynamic)?\r\n override var
onended: ((Event) -> dynamic)?\r\n override var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?\r\n
override var onfocus: ((FocusEvent) -> dynamic)?\r\n override var oninput: ((InputEvent) -> dynamic)?\r\n
override var oninvalid: ((Event) -> dynamic)?\r\n override var onkeydown: ((KeyboardEvent) -> dynamic)?\r\n
override var onkeypress: ((KeyboardEvent) -> dynamic)?\r\n override var onkeyup: ((KeyboardEvent) ->
dynamic)?\r\n override var onload: ((Event) -> dynamic)?\r\n override var onloadeddata: ((Event) ->
dynamic)?\r\n override var onloadedmetadata: ((Event) -> dynamic)?\r\n override var onloadend: ((Event) ->
dynamic)?\r\n override var onloadstart: ((ProgressEvent) -> dynamic)?\r\n override var onmousedown:
((MouseEvent) -> dynamic)?\r\n override var onmouseenter: ((MouseEvent) -> dynamic)?\r\n override var
onmouseleave: ((MouseEvent) -> dynamic)?\r\n override var onmousemove: ((MouseEvent) -> dynamic)?\r\n
override var onmouseout: ((MouseEvent) -> dynamic)?\r\n override var onmouseover: ((MouseEvent) ->
dynamic)?\r\n override var onmouseup: ((MouseEvent) -> dynamic)?\r\n override var onwheel: ((WheelEvent) -
> dynamic)?\r\n override var onpause: ((Event) -> dynamic)?\r\n override var onplay: ((Event) -> dynamic)?\r\n
override var onplaying: ((Event) -> dynamic)?\r\n override var onprogress: ((ProgressEvent) -> dynamic)?\r\n
override var onratechange: ((Event) -> dynamic)?\r\n override var onreset: ((Event) -> dynamic)?\r\n override
var onresize: ((Event) -> dynamic)?\r\n override var onscroll: ((Event) -> dynamic)?\r\n override var onseeked:
((Event) -> dynamic)?\r\n override var onseeking: ((Event) -> dynamic)?\r\n override var onselect: ((Event) ->
dynamic)?\r\n override var onshow: ((Event) -> dynamic)?\r\n override var onstalled: ((Event) -> dynamic)?\r\n
override var onsubmit: ((Event) -> dynamic)?\r\n override var onsuspend: ((Event) -> dynamic)?\r\n override var
ontimeupdate: ((Event) -> dynamic)?\r\n override var ontoggle: ((Event) -> dynamic)?\r\n override var
onvolumechange: ((Event) -> dynamic)?\r\n override var onwaiting: ((Event) -> dynamic)?\r\n override var
oncopy: ((ClipboardEvent) -> dynamic)?\r\n override var oncut: ((ClipboardEvent) -> dynamic)?\r\n override
var onpaste: ((ClipboardEvent) -> dynamic)?\r\n override val fullscreenElement: Element?\r\n override val
children: HTMLCollection\r\n override val firstElementChild: Element?\r\n override val lastElementChild:
Element?\r\n override val childElementCount: Int\r\n fun exitFullscreen(): Promise<Unit>\r\n fun
getElementsByName(elementName: String): NodeList\r\n fun open(type: String = definedExternally, replace:
String = definedExternally): Document\r\n fun open(url: String, name: String, features: String): Window\r\n fun
close(): Unit\r\n fun write(vararg text: String): Unit\r\n fun writeln(vararg text: String): Unit\r\n fun
hasFocus(): Boolean\r\n fun execCommand(commandId: String, showUI: Boolean = definedExternally, value:
String = definedExternally): Boolean\r\n fun queryCommandEnabled(commandId: String): Boolean\r\n fun
queryCommandIndeterm(commandId: String): Boolean\r\n fun queryCommandState(commandId: String):
Boolean\r\n fun queryCommandSupported(commandId: String): Boolean\r\n fun
queryCommandValue(commandId: String): String\r\n fun clear(): Unit\r\n fun captureEvents(): Unit\r\n fun
releaseEvents(): Unit\r\n fun getElementsByTagName(qualifiedName: String): HTMLCollection\r\n fun
getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection\r\n fun
getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection\r\n fun
createElement(localName: String,
options: ElementCreationOptions = definedExternally): Element\r\n fun createElementNS(namespace: String?,
qualifiedName: String, options: ElementCreationOptions = definedExternally): Element\r\n fun
createDocumentFragment(): DocumentFragment\r\n fun createTextNode(data: String): Text\r\n fun
createCDATASection(data: String): CDATASection\r\n fun createComment(data: String): Comment\r\n fun
createProcessingInstruction(target: String, data: String): ProcessingInstruction\r\n fun importNode(node: Node,
deep: Boolean = definedExternally): Node\r\n fun adoptNode(node: Node): Node\r\n fun

```



```

definedExternally): UnionElementOrHTMLCollection? { } @kotlin.internal.InlineOnly inline operator fun
HTMLAllCollection.get(index: Int): Element? = asDynamic()[index] @kotlin.internal.InlineOnly inline
operator fun HTMLAllCollection.get(name: String): UnionElementOrHTMLCollection? =
asDynamic()[name] @kotlin.internal.InlineOnly * Exposes the JavaScript
[HTMLFormControlsCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLFormControlsCollection)
to Kotlin @kotlin.internal.InlineOnly public external abstract class HTMLFormControlsCollection : HTMLCollection {
 override fun namedItem(name: String): UnionElementOrRadioNodeList? { } @kotlin.internal.InlineOnly override
inline operator fun HTMLFormControlsCollection.get(name: String): UnionElementOrRadioNodeList? =
asDynamic()[name] @kotlin.internal.InlineOnly * Exposes the JavaScript
[RadioNodeList](https://developer.mozilla.org/en/docs/Web/API/RadioNodeList) to Kotlin @kotlin.internal.InlineOnly public external
abstract class RadioNodeList : NodeList, UnionElementOrRadioNodeList {
 open var value: String { } @kotlin.internal.InlineOnly * Exposes the JavaScript
[HTMLOptionsCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLOptionsCollection) to Kotlin @kotlin.internal.InlineOnly public external
abstract class HTMLOptionsCollection : HTMLCollection {
 override var length: Int
 open var selectedIndex: Int
 fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally): Unit
 fun remove(index: Int): Unit { } @kotlin.internal.InlineOnly inline
operator fun HTMLOptionsCollection.set(index: Int, option: HTMLOptionElement?): Unit {
 asDynamic()[index] = option; } @kotlin.internal.InlineOnly * Exposes the JavaScript
[HTMLInputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLInputElement) to Kotlin @kotlin.internal.InlineOnly public external
abstract class HTMLInputElement : Element, ElementCSSInlineStyle, GlobalEventHandlers,
DocumentAndElementEventHandlers, ElementContentEditable {
 open var title: String
 open var lang: String
 open var translate: Boolean
 open var dir: String
 open val dataset: DOMStringMap
 open var hidden: Boolean
 open var tabIndex: Int
 open var accessKey: String
 open val accessKeyLabel: String
 open var draggable: Boolean
 open val dropzone: DOMTokenList
 open var contextMenu: HTMLMenuElement?
 open var spellcheck: Boolean
 open var innerText: String
 open val offsetParent: Element?
 open val offsetTop: Int
 open val offsetLeft: Int
 open val offsetWidth: Int
 open val offsetHeight: Int
 fun click(): Unit
 fun focus(): Unit
 fun blur(): Unit
 fun forceSpellCheck(): Unit { } @kotlin.internal.InlineOnly * Exposes the JavaScript
[HTMLUnknownElement](https://developer.mozilla.org/en/docs/Web/API/HTMLUnknownElement) to Kotlin @kotlin.internal.InlineOnly public external
abstract class HTMLUnknownElement : HTMLInputElement { } @kotlin.internal.InlineOnly * Exposes the
JavaScript [DOMStringMap](https://developer.mozilla.org/en/docs/Web/API/DOMStringMap) to Kotlin @kotlin.internal.InlineOnly public external
abstract class DOMStringMap {
 @kotlin.internal.InlineOnly inline operator fun
DOMStringMap.get(name: String): String? = asDynamic()[name] @kotlin.internal.InlineOnly inline operator fun
DOMStringMap.set(name: String, value: String): Unit {
 asDynamic()[name] = value; } @kotlin.internal.InlineOnly * Exposes the
JavaScript [HTMLHtmlElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHtmlElement) to Kotlin @kotlin.internal.InlineOnly public external
abstract class HTMLHtmlElement : HTMLInputElement {
 open var version: String { } @kotlin.internal.InlineOnly * Exposes the JavaScript
[HTMLHeadElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadElement) to Kotlin @kotlin.internal.InlineOnly public external
abstract class HTMLHeadElement : HTMLInputElement { } @kotlin.internal.InlineOnly * Exposes the
JavaScript [HTMLTitleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTitleElement) to Kotlin @kotlin.internal.InlineOnly public external
abstract class HTMLTitleElement : HTMLInputElement {
 open var text: String { } @kotlin.internal.InlineOnly * Exposes the JavaScript
[HTMLBaseElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBaseElement) to Kotlin @kotlin.internal.InlineOnly public external
abstract class HTMLBaseElement : HTMLInputElement {
 open var href: String
 open var target: String { } @kotlin.internal.InlineOnly * Exposes the JavaScript
[HTMLLinkElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLinkElement) to Kotlin @kotlin.internal.InlineOnly public external
abstract class HTMLLinkElement : HTMLInputElement, LinkStyle {
 open var scope: String
 open var workerType: WorkerType
 open var href: String
 open var crossOrigin: String? { }

```



```

open var rel: String\r\n @JsName("\as") open var as_: RequestDestination\r\n open val relList:
DOMTokenList\r\n open var media: String\r\n open var nonce: String\r\n open var hreflang: String\r\n open
var type: String\r\n open val sizes: DOMTokenList\r\n open var referrerPolicy: String\r\n open var charset:
String\r\n open var rev: String\r\n open var target: String\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLMetaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMetaElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLMetaElement : HTMLElement {\r\n open var name: String\r\n open
var httpEquiv: String\r\n open var content: String\r\n open var scheme: String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the
JavaScript [HTMLStyleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLStyleElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLStyleElement : HTMLElement, LinkStyle {\r\n open var media:
String\r\n open var nonce: String\r\n open var type: String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLBodyElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBodyElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLBodyElement : HTMLElement, WindowEventHandlers {\r\n open var
text: String\r\n open var link: String\r\n open var vLink: String\r\n open var aLink: String\r\n open var
bgColor: String\r\n open var background: String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLHeadingElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadingElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLHeadingElement : HTMLElement {\r\n open var align:
String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLParagraphElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParagraphElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLParagraphElement : HTMLElement {\r\n open var align:
String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLHRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHRElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLHRElement : HTMLElement {\r\n open var align: String\r\n open var color:
String\r\n open var noShade: Boolean\r\n open var size: String\r\n open var width: String\r\n}\r\n\r\n\r\n/**\r\n *
Exposes the JavaScript [HTMLPreElement](https://developer.mozilla.org/en/docs/Web/API/HTMLPreElement) to
Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLPreElement : HTMLElement {\r\n open var width:
Int\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLQuoteElement](https://developer.mozilla.org/en/docs/Web/API/HTMLQuoteElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLQuoteElement : HTMLElement {\r\n open var cite:
String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLLOListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLOListElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLLOListElement : HTMLElement {\r\n open var reversed: Boolean\r\n
open var start: Int\r\n open var type: String\r\n open var compact: Boolean\r\n}\r\n\r\n\r\n/**\r\n * Exposes the
JavaScript [HTMLULListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLULListElement) to
Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLULListElement : HTMLElement {\r\n open var compact:
Boolean\r\n open var type: String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLLIElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLIElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLLIElement : HTMLElement {\r\n open var value: Int\r\n open var type:
String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLDListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDListElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLDListElement : HTMLElement {\r\n open var compact:
Boolean\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLDivElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDivElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLDivElement : HTMLElement {\r\n open var align: String\r\n}\r\n\r\n\r\n
Exposes the JavaScript
[HTMLAnchorElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAnchorElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLAnchorElement : HTMLElement, HTMLHyperlinkElementUtils {\r\n
open var target: String\r\n open var download: String\r\n open var ping: String\r\n open var rel: String\r\n
open val relList: DOMTokenList\r\n open var hreflang: String\r\n open var type: String\r\n open var text:

```

```

String\r\n open var referrerPolicy: String\r\n open var coords: String\r\n open var charset: String\r\n open var
name: String\r\n open var rev: String\r\n open var shape: String\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLDataElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataElement) to Kotlin\r\n
*/\r\n\r\npublic external abstract class HTMLDataElement : HTMLInputElement {\r\n open var value:
String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLTimeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTimeElement) to Kotlin\r\n
*/\r\n\r\npublic external abstract class HTMLTimeElement : HTMLInputElement {\r\n open var dateTime:
String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLSpanElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSpanElement) to Kotlin\r\n
*/\r\n\r\npublic external abstract class HTMLSpanElement : HTMLInputElement {\r\n}\r\n\r\n\r\n/**\r\n * Exposes the
JavaScript [HTMLBRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBRElement) to Kotlin\r\n
*/\r\n\r\npublic external abstract class HTMLBRElement : HTMLInputElement {\r\n open var clear:
String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLHyperlinkElementUtils](https://developer.mozilla.org/en/docs/Web/API/HTMLHyperlinkElementUtils) to
Kotlin\r\n
*/\r\n\r\npublic external interface HTMLHyperlinkElementUtils {\r\n var href: String\r\n val origin:
String\r\n var protocol: String\r\n var username: String\r\n var password: String\r\n var host: String\r\n var
hostname: String\r\n var port: String\r\n var pathname: String\r\n var search: String\r\n var hash:
String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLModElement](https://developer.mozilla.org/en/docs/Web/API/HTMLModElement) to Kotlin\r\n
*/\r\n\r\npublic external abstract class HTMLModElement : HTMLInputElement {\r\n open var cite: String\r\n open var
dateTime: String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLPictureElement](https://developer.mozilla.org/en/docs/Web/API/HTMLPictureElement) to Kotlin\r\n
*/\r\n\r\npublic external abstract class HTMLPictureElement : HTMLInputElement {\r\n}\r\n\r\n\r\n/**\r\n * Exposes the
JavaScript [HTMLSourceElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSourceElement) to
Kotlin\r\n
*/\r\n\r\npublic external abstract class HTMLSourceElement : HTMLInputElement {\r\n open var src: String\r\n open
var type: String\r\n open var srcset: String\r\n open var sizes: String\r\n open var media:
String\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLImageElement](https://developer.mozilla.org/en/docs/Web/API/HTMLImageElement) to Kotlin\r\n
*/\r\n\r\npublic external abstract class HTMLImageElement : HTMLInputElement, TexImageSource,
HTMLOrSVGImageElement {\r\n open var alt: String\r\n open var src: String\r\n open var srcset: String\r\n open
var sizes: String\r\n open var crossOrigin: String?\r\n open var useMap: String\r\n open var isMap:
Boolean\r\n open var width: Int\r\n open var height: Int\r\n open val naturalWidth: Int\r\n open val
naturalHeight: Int\r\n open val complete: Boolean\r\n open val currentSrc: String\r\n open var referrerPolicy:
String\r\n open var name: String\r\n open var lowsrc: String\r\n open var align: String\r\n open var hspace:
Int\r\n open var vspace: Int\r\n open var longDesc: String\r\n open var border: String\r\n open val x: Int\r\n open
val y: Int\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[Image](https://developer.mozilla.org/en/docs/Web/API/Image) to Kotlin\r\n
*/\r\n\r\npublic external open class
Image(width: Int = definedExternally, height: Int = definedExternally) : HTMLImageElement {\r\n override val
children: HTMLCollection\r\n override val firstElementChild: Element?\r\n override val lastElementChild:
Element?\r\n override val childElementCount: Int\r\n override val previousElementSibling: Element?\r\n override
val nextElementSibling: Element?\r\n override val assignedSlot: HTMLSlotElement?\r\n override val style:
CSSStyleDeclaration\r\n override var ongotpointercapture: ((PointerEvent) -> dynamic)?\r\n override var
onlostpointercapture: ((PointerEvent) -> dynamic)?\r\n override var onpointerdown: ((PointerEvent) ->
dynamic)?\r\n override var onpointermove: ((PointerEvent) -> dynamic)?\r\n override var onpointerup:
((PointerEvent) -> dynamic)?\r\n override var onpointercancel: ((PointerEvent) -> dynamic)?\r\n override var
onpointerover: ((PointerEvent) -> dynamic)?\r\n override var onpointerout: ((PointerEvent) -> dynamic)?\r\n
 override var onpointerenter: ((PointerEvent) -> dynamic)?\r\n override var onpointerleave: ((PointerEvent) ->
dynamic)?\r\n override var onabort: ((Event) -> dynamic)?\r\n override var onblur: ((FocusEvent) ->

```

```

dynamic)?\r\n override var onCancel: ((Event) -> dynamic)?\r\n override var oncanplay: ((Event) ->
dynamic)?\r\n override var oncanplaythrough: ((Event) -> dynamic)?\r\n override var onChange: ((Event) ->
dynamic)?\r\n override var onclick: ((MouseEvent) -> dynamic)?\r\n override var onClose: ((Event) ->
dynamic)?\r\n override var oncontextmenu: ((MouseEvent) -> dynamic)?\r\n override var oncuechange: ((Event)
-> dynamic)?\r\n override var ondblclick: ((MouseEvent) -> dynamic)?\r\n override var ondrag: ((DragEvent) ->
dynamic)?\r\n override var ondragend: ((DragEvent) -> dynamic)?\r\n override var ondragenter: ((DragEvent) ->
dynamic)?\r\n override var ondragexit: ((DragEvent) -> dynamic)?\r\n override var ondragleave: ((DragEvent) -
> dynamic)?\r\n override var ondragover: ((DragEvent) -> dynamic)?\r\n override var ondragstart: ((DragEvent)
-> dynamic)?\r\n override var ondrop: ((DragEvent) -> dynamic)?\r\n override var ondurationchange: ((Event) -
> dynamic)?\r\n override var onemptied: ((Event) -> dynamic)?\r\n override var onended: ((Event) ->
dynamic)?\r\n override var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?\r\n override var onFocus:
((FocusEvent) -> dynamic)?\r\n override var oninput: ((InputEvent) -> dynamic)?\r\n override var oninvalid:
((Event) -> dynamic)?\r\n override var onkeydown: ((KeyboardEvent) -> dynamic)?\r\n override var
onkeypress: ((KeyboardEvent) -> dynamic)?\r\n override var onkeyup: ((KeyboardEvent) -> dynamic)?\r\n
override var onload: ((Event) -> dynamic)?\r\n override var onloadeddata: ((Event) -> dynamic)?\r\n override
var onloadedmetadata: ((Event) -> dynamic)?\r\n override var onloadend: ((Event) -> dynamic)?\r\n override var
onloadstart: ((ProgressEvent) -> dynamic)?\r\n override var onmousedown: ((MouseEvent) -> dynamic)?\r\n
override var onmouseenter: ((MouseEvent) -> dynamic)?\r\n override var onmouseleave: ((MouseEvent) ->
dynamic)?\r\n override var onmousemove: ((MouseEvent) -> dynamic)?\r\n override var onmouseout:
((MouseEvent) -> dynamic)?\r\n override var onmouseover: ((MouseEvent) -> dynamic)?\r\n override var
onmouseup: ((MouseEvent) -> dynamic)?\r\n override var onwheel: ((WheelEvent) -> dynamic)?\r\n override
var onpause: ((Event) -> dynamic)?\r\n override var onplay: ((Event) -> dynamic)?\r\n override var onplaying:
((Event) -> dynamic)?\r\n override var onprogress: ((ProgressEvent) -> dynamic)?\r\n override var
onratechange: ((Event) -> dynamic)?\r\n override var onreset: ((Event) -> dynamic)?\r\n override var onresize:
((Event) -> dynamic)?\r\n override var onscroll: ((Event) -> dynamic)?\r\n override var onseeked: ((Event) ->
dynamic)?\r\n override var onseeking: ((Event) -> dynamic)?\r\n override var onselect: ((Event) ->
dynamic)?\r\n override var onshow: ((Event) -> dynamic)?\r\n override var onstalled: ((Event) -> dynamic)?\r\n
override var onsubmit: ((Event) -> dynamic)?\r\n override var onsuspend: ((Event) -> dynamic)?\r\n override var
ontimeupdate: ((Event) -> dynamic)?\r\n override var ontoggle: ((Event) -> dynamic)?\r\n override var
onvolumechange: ((Event) -> dynamic)?\r\n override var onwaiting: ((Event) -> dynamic)?\r\n override var
oncopy: ((ClipboardEvent) -> dynamic)?\r\n override var oncut: ((ClipboardEvent) -> dynamic)?\r\n override
var onpaste: ((ClipboardEvent) -> dynamic)?\r\n override var contentEditable: String\r\n override val
isContentEditable: Boolean\r\n override fun prepend(vararg nodes: dynamic): Unit\r\n override fun
append(vararg nodes: dynamic): Unit\r\n override fun querySelector(selectors: String): Element?\r\n override
fun querySelectorAll(selectors: String): NodeList\r\n override fun before(vararg nodes: dynamic): Unit\r\n
override fun after(vararg nodes: dynamic): Unit\r\n override fun replaceWith(vararg nodes: dynamic): Unit\r\n
override fun remove(): Unit\r\n override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */):
Array<DOMQuad>\r\n override fun convertQuadFromNode(quad: dynamic, from: dynamic, options:
ConvertCoordinateOptions /* = definedExternally */): DOMQuad\r\n override fun convertRectFromNode(rect:
DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\r\n
override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* =
definedExternally */): DOMPoint\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLIFrameElement](https://developer.mozilla.org/en/docs/Web/API/HTMLIFrameElement) to Kotlin\r\n
*/\r\npublic external abstract class HTMLIFrameElement : HTMLElement {\r\n open var src: String\r\n open var
srcdoc: String\r\n open var name: String\r\n open val sandbox: DOMTokenList\r\n open var allowFullscreen:
Boolean\r\n open var allowUserMedia: Boolean\r\n open var width: String\r\n open var height: String\r\n
open var referrerPolicy: String\r\n open val contentDocument: Document?\r\n open val contentWindow:
Window?\r\n open var align: String\r\n open var scrolling: String\r\n open var frameborder: String\r\n open

```



```

((FocusEvent) -> dynamic)?\r\n override var oninput: ((InputEvent) -> dynamic)?\r\n override var oninvalid:
((Event) -> dynamic)?\r\n override var onkeydown: ((KeyboardEvent) -> dynamic)?\r\n override var
onkeypress: ((KeyboardEvent) -> dynamic)?\r\n override var onkeyup: ((KeyboardEvent) -> dynamic)?\r\n
override var onload: ((Event) -> dynamic)?\r\n override var onloadeddata: ((Event) -> dynamic)?\r\n override
var onloadedmetadata: ((Event) -> dynamic)?\r\n override var onloadend: ((Event) -> dynamic)?\r\n override var
onloadstart: ((ProgressEvent) -> dynamic)?\r\n override var onmousedown: ((MouseEvent) -> dynamic)?\r\n
override var onmouseenter: ((MouseEvent) -> dynamic)?\r\n override var onmouseleave: ((MouseEvent) ->
dynamic)?\r\n override var onmousemove: ((MouseEvent) -> dynamic)?\r\n override var onmouseout:
((MouseEvent) -> dynamic)?\r\n override var onmouseover: ((MouseEvent) -> dynamic)?\r\n override var
onmouseup: ((MouseEvent) -> dynamic)?\r\n override var onwheel: ((WheelEvent) -> dynamic)?\r\n override
var onpause: ((Event) -> dynamic)?\r\n override var onplay: ((Event) -> dynamic)?\r\n override var onplaying:
((Event) -> dynamic)?\r\n override var onprogress: ((ProgressEvent) -> dynamic)?\r\n override var
onratechange: ((Event) -> dynamic)?\r\n override var onreset: ((Event) -> dynamic)?\r\n override var onresize:
((Event) -> dynamic)?\r\n override var onscroll: ((Event) -> dynamic)?\r\n override var onseeked: ((Event) ->
dynamic)?\r\n override var onseeking: ((Event) -> dynamic)?\r\n override var onselect: ((Event) ->
dynamic)?\r\n override var onshow: ((Event) -> dynamic)?\r\n override var onstalled: ((Event) -> dynamic)?\r\n
override var onsubmit: ((Event) -> dynamic)?\r\n override var onsuspend: ((Event) -> dynamic)?\r\n override var
ontimeupdate: ((Event) -> dynamic)?\r\n override var ontoggle: ((Event) -> dynamic)?\r\n override var
onvolumechange: ((Event) -> dynamic)?\r\n override var onwaiting: ((Event) -> dynamic)?\r\n override var
oncopy: ((ClipboardEvent) -> dynamic)?\r\n override var oncut: ((ClipboardEvent) -> dynamic)?\r\n override
var onpaste: ((ClipboardEvent) -> dynamic)?\r\n override var contentEditable: String\r\n override val
isContentEditable: Boolean\r\n override fun prepend(vararg nodes: dynamic): Unit\r\n override fun
append(vararg nodes: dynamic): Unit\r\n override fun querySelector(selectors: String): Element?\r\n override
fun querySelectorAll(selectors: String): NodeList\r\n override fun before(vararg nodes: dynamic): Unit\r\n
override fun after(vararg nodes: dynamic): Unit\r\n override fun replaceWith(vararg nodes: dynamic): Unit\r\n
override fun remove(): Unit\r\n override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */):
Array<DOMQuad>\r\n override fun convertQuadFromNode(quad: dynamic, from: dynamic, options:
ConvertCoordinateOptions /* = definedExternally */): DOMQuad\r\n override fun convertRectFromNode(rect:
DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\r\n
override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* =
definedExternally */): DOMPoint\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLTrackElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTrackElement) to Kotlin\r\n
*/\r\npublic external abstract class HTMLTrackElement : HTMLElement {\r\n open var kind: String\r\n open
var src: String\r\n open var srclang: String\r\n open var label: String\r\n open var default: Boolean\r\n open
val readyState: Short\r\n open val track: TextTrack\r\n\r\n companion object {\r\n val NONE: Short\r\n
val LOADING: Short\r\n val LOADED: Short\r\n val ERROR: Short\r\n }\r\n}\r\n\r\n/**\r\n * Exposes
the JavaScript [HTMLMediaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMediaElement) to
Kotlin\r\n
*/\r\npublic external abstract class HTMLMediaElement : HTMLElement {\r\n open val error:
MediaError?\r\n open var src: String\r\n open var srcObject: dynamic\r\n open val currentSrc: String\r\n open
var crossOrigin: String?\r\n open val networkState: Short\r\n open var preload: String\r\n open val buffered:
TimeRanges\r\n open val readyState: Short\r\n open val seeking: Boolean\r\n open var currentTime:
Double\r\n open val duration: Double\r\n open val paused: Boolean\r\n open var defaultPlaybackRate:
Double\r\n open var playbackRate: Double\r\n open val played: TimeRanges\r\n open val seekable:
TimeRanges\r\n open val ended: Boolean\r\n open var autoplay: Boolean\r\n open var loop: Boolean\r\n
open var controls: Boolean\r\n open var volume: Double\r\n open var muted: Boolean\r\n open var
defaultMuted: Boolean\r\n open val audioTracks: AudioTrackList\r\n open val videoTracks: VideoTrackList\r\n
open val textTracks: TextTrackList\r\n fun load(): Unit\r\n fun canPlayType(type: String):
CanPlayTypeResult\r\n fun fastSeek(time: Double): Unit\r\n fun getStartDate(): dynamic\r\n fun play():

```



```

abstract class TimeRanges {
 open val length: Int
 fun start(index: Int): Double
 fun end(index: Int): Double
}
Exposes the JavaScript
[TrackEvent](https://developer.mozilla.org/en/docs/Web/API/TrackEvent) to Kotlin
public external open class TrackEvent(
 type: String, eventInitDict: TrackEventInit = definedExternally
) : Event {
 open val track: UnionAudioTrackOrTextTrackOrVideoTrack?
}
public external interface TrackEventInit : EventInit {
 var track: UnionAudioTrackOrTextTrackOrVideoTrack? /* = null */
 get() = definedExternally
 set(value) = definedExternally
}
@kotlin.internal.InlineOnly
public inline fun TrackEventInit(
 track: UnionAudioTrackOrTextTrackOrVideoTrack? = null,
 bubbles: Boolean? = false,
 cancelable: Boolean? = false,
 composed: Boolean? = false
): TrackEventInit {
 val o = js("{}")
 o["track"] = track
 o["bubbles"] = bubbles
 o["cancelable"] = cancelable
 o["composed"] = composed
}
return o
Exposes the JavaScript
[HTMLMapElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMapElement) to Kotlin
public external abstract class HTMLMapElement : HTMLElement {
 open var name: String
 open val areas: HTMLCollection
}
Exposes the JavaScript
[HTMLAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAreaElement) to Kotlin
public external abstract class HTMLAreaElement : HTMLElement, HTMLHyperlinkElementUtils {
 open var alt: String
 open var coords: String
 open var shape: String
 open var target: String
 open var download: String
 open var ping: String
 open var rel: String
 open val relList: DOMTokenList
 open var referrerPolicy: String
 open var noHref: Boolean
}
Exposes the JavaScript
[HTMLTableElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableElement) to Kotlin
public external abstract class HTMLTableElement : HTMLElement {
 open var caption: HTMLTableCaptionElement?
 open var tHead: HTMLTableSectionElement?
 open var tFoot: HTMLTableSectionElement?
 open val tBodies: HTMLCollection
 open val rows: HTMLCollection
 open var align: String
 open var border: String
 open var frame: String
 open var rules: String
 open var summary: String
 open var width: String
 open var bgColor: String
 open var cellPadding: String
 open var cellSpacing: String
 fun createCaption(): HTMLTableCaptionElement
 fun deleteCaption(): Unit
 fun createTHead(): HTMLTableSectionElement
 fun deleteTHead(): Unit
 fun createTFoot(): HTMLTableSectionElement
 fun deleteTFoot(): Unit
 fun createTBody(): HTMLTableSectionElement
 fun insertRow(index: Int = definedExternally): HTMLTableRowElement
 fun deleteRow(index: Int): Unit
}
Exposes the JavaScript
[HTMLTableCaptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCaptionElement) to Kotlin
public external abstract class HTMLTableCaptionElement : HTMLElement {
 open var align: String
}
Exposes the JavaScript
[HTMLTableColElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableColElement) to Kotlin
public external abstract class HTMLTableColElement : HTMLElement {
 open var span: Int
 open var align: String
 open var ch: String
 open var chOff: String
 open var vAlign: String
 open var width: String
}
Exposes the JavaScript
[HTMLTableSectionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableSectionElement) to Kotlin
public external abstract class HTMLTableSectionElement : HTMLElement {
 open val rows: HTMLCollection
 open var align: String
 open var ch: String
 open var chOff: String
 open var vAlign: String
 fun insertRow(index: Int = definedExternally): HTMLElement
 fun deleteRow(index: Int): Unit
}
Exposes the JavaScript
[HTMLTableRowElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableRowElement) to Kotlin
public external abstract class HTMLTableRowElement : HTMLElement {
 open val rowIndex: Int
 open val sectionRowIndex: Int
 open val cells: HTMLCollection
 open var align: String
 open var ch: String
 open var chOff: String
 open var vAlign: String
 open var bgColor: String
 fun insertCell(index: Int = definedExternally): HTMLElement
 fun deleteCell(index: Int): Unit
}
Exposes the JavaScript

```

[HTMLTableCellElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCellElement) to Kotlin

```

*\r\npublic external abstract class HTMLTableCellElement : HTMLElement {\r\n
 open var colSpan: Int\r\n
 open var rowSpan: Int\r\n
 open var headers: String\r\n
 open val cellIndex: Int\r\n
 open var scope: String\r\n
 open var abbr: String\r\n
 open var align: String\r\n
 open var axis: String\r\n
 open var height: String\r\n
 open var width: String\r\n
 open var ch: String\r\n
 open var chOff: String\r\n
 open var noWrap: Boolean\r\n
 open var vAlign: String\r\n
 open var bgColor: String\r\n}\r\n\r\n**\r\n * Exposes the JavaScript

```

[HTMLFormElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFormElement) to Kotlin

```

*\r\npublic external abstract class HTMLFormElement : HTMLElement {\r\n
 open var acceptCharset: String\r\n
 open var action: String\r\n
 open var autocomplete: String\r\n
 open var enctype: String\r\n
 open var encoding: String\r\n
 open var method: String\r\n
 open var name: String\r\n
 open var noValidate: Boolean\r\n
 open var target: String\r\n
 open val elements: HTMLFormControlsCollection\r\n
 open val length: Int\r\n
 fun submit(): Unit\r\n
 fun reset(): Unit\r\n
 fun checkValidity(): Boolean\r\n
 fun reportValidity(): Boolean\r\n}
\r\n@kotlin.internal.InlineOnly inline operator fun HTMLFormElement.get(index: Int): Element? =
asDynamic()[index]\r\n@kotlin.internal.InlineOnly inline operator fun HTMLFormElement.get(name: String):
UnionElementOrRadioNodeList? = asDynamic()[name]\r\n\r\n**\r\n * Exposes the JavaScript

```

[HTMLLabelElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLabelElement) to Kotlin

```

*\r\npublic external abstract class HTMLLabelElement : HTMLElement {\r\n
 open val form: HTMLFormElement?\r\n
 open var htmlFor: String\r\n
 open val control: HTMLElement?\r\n}\r\n\r\n**\r\n *
Exposes the JavaScript

```

[HTMLInputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLInputElement) to Kotlin

```

*\r\npublic external abstract class HTMLInputElement : HTMLElement {\r\n
 open var accept: String\r\n
 open var alt: String\r\n
 open var autocomplete: String\r\n
 open var autofocus: Boolean\r\n
 open var defaultChecked: Boolean\r\n
 open var checked: Boolean\r\n
 open var dirName: String\r\n
 open var disabled: Boolean\r\n
 open val form: HTMLFormElement?\r\n
 open val files: FileList?\r\n
 open var formAction: String\r\n
 open var formEnctype: String\r\n
 open var formMethod: String\r\n
 open var formNoValidate: Boolean\r\n
 open var formTarget: String\r\n
 open var height: Int\r\n
 open var indeterminate: Boolean\r\n
 open var inputMode: String\r\n
 open val list: HTMLElement?\r\n
 open var max: String\r\n
 open var maxLength: Int\r\n
 open var min: String\r\n
 open var minLength: Int\r\n
 open var multiple: Boolean\r\n
 open var name: String\r\n
 open var pattern: String\r\n
 open var placeholder: String\r\n
 open var readOnly: Boolean\r\n
 open var required: Boolean\r\n
 open var size: Int\r\n
 open var src: String\r\n
 open var step: String\r\n
 open var type: String\r\n
 open var defaultValue: String\r\n
 open var value: String\r\n
 open var valueAsDate: dynamic\r\n
 open var valueAsNumber: Double\r\n
 open var width: Int\r\n
 open val willValidate: Boolean\r\n
 open val validity: ValidityState\r\n
 open val validationMessage: String\r\n
 open val labels: NodeList\r\n
 open var selectionStart: Int?\r\n
 open var selectionEnd: Int?\r\n
 open var selectionDirection: String?\r\n
 open var align: String\r\n
 open var useMap: String\r\n
 fun stepUp(n: Int = definedExternally): Unit\r\n
 fun stepDown(n: Int = definedExternally): Unit\r\n
 fun checkValidity(): Boolean\r\n
 fun reportValidity(): Boolean\r\n
 fun setCustomValidity(error: String): Unit\r\n
 fun select(): Unit\r\n
 fun setRangeText(replacement: String): Unit\r\n
 fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally): Unit\r\n
 fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally): Unit\r\n}\r\n\r\n**\r\n * Exposes the JavaScript

```

[HTMLButtonElement](https://developer.mozilla.org/en/docs/Web/API/HTMLButtonElement) to Kotlin

```

*\r\npublic external abstract class HTMLButtonElement : HTMLElement {\r\n
 open var autofocus: Boolean\r\n
 open var disabled: Boolean\r\n
 open val form: HTMLFormElement?\r\n
 open var formAction: String\r\n
 open var formEnctype: String\r\n
 open var formMethod: String\r\n
 open var formNoValidate: Boolean\r\n
 open var formTarget: String\r\n
 open var name: String\r\n
 open var type: String\r\n
 open var value: String\r\n
 open var menu: HTMLMenuElement?\r\n
 open val willValidate: Boolean\r\n
 open val validity: ValidityState\r\n
 open val validationMessage: String\r\n
 open val labels: NodeList\r\n
 fun checkValidity(): Boolean\r\n
 fun reportValidity(): Boolean\r\n
 fun setCustomValidity(error: String): Unit\r\n}\r\n\r\n**\r\n * Exposes the JavaScript

```

[HTMLSelectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSelectElement) to



```

Kotlin\n\n *\n\npublic external abstract class HTMLSelectElement : HTMLInputElement, ItemArrayLike<Element> {\n\n open var autocomplete: String\n open var autofocus: Boolean\n open var disabled: Boolean\n open val form: HTMLFormElement?\n open var multiple: Boolean\n open var name: String\n open var required: Boolean\n open var size: Int\n open val type: String\n open val options: HTMLOptionsCollection\n override var length: Int\n open val selectedOptions: HTMLCollection\n open var selectedIndex: Int\n open var value: String\n open val willValidate: Boolean\n open val validity: ValidityState\n open val validationMessage: String\n open val labels: NodeList\n override fun item(index: Int): Element?\n fun namedItem(name: String): HTMLOptionElement?\n fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally): Unit\n fun remove(index: Int): Unit\n fun checkValidity(): Boolean\n fun reportValidity(): Boolean\n fun setCustomValidity(error: String): Unit\n}\n\n@kotlin.internal.InlineOnly inline operator fun HTMLSelectElement.get(index: Int): Element? = asDynamic()[index]\n\n@kotlin.internal.InlineOnly inline operator fun HTMLSelectElement.set(index: Int, option: HTMLOptionElement?): Unit { asDynamic()[index] = option; }\n\n*\n\n * Exposes the JavaScript [HTMLDataListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataListElement) to Kotlin\n\n *\n\npublic external abstract class HTMLDataListElement : HTMLInputElement {\n open val options: HTMLCollection\n}\n\n*\n\n * Exposes the JavaScript [HTMLOptGroupElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptGroupElement) to Kotlin\n\n *\n\npublic external abstract class HTMLOptGroupElement : HTMLInputElement, UnionHTMLOptGroupElementOrHTMLOptionElement {\n open var disabled: Boolean\n open var label: String\n}\n\n*\n\n * Exposes the JavaScript [HTMLOptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptionElement) to Kotlin\n\n *\n\npublic external abstract class HTMLOptionElement : HTMLInputElement, UnionHTMLOptGroupElementOrHTMLOptionElement {\n open var disabled: Boolean\n open val form: HTMLFormElement?\n open var label: String\n open var defaultSelected: Boolean\n open var selected: Boolean\n open var value: String\n open var text: String\n open val index: Int\n}\n\n*\n\n * Exposes the JavaScript [Option](https://developer.mozilla.org/en/docs/Web/API/Option) to Kotlin\n\n *\n\npublic external open class Option(text: String = definedExternally, value: String = definedExternally, defaultSelected: Boolean = definedExternally, selected: Boolean = definedExternally) : HTMLOptionElement {\n override val children: HTMLCollection\n override val firstElementChild: Element?\n override val lastElementChild: Element?\n override val childElementCount: Int\n override val previousElementSibling: Element?\n override val nextElementSibling: Element?\n override val assignedSlot: HTMLSlotElement?\n override val style: CSSStyleDeclaration\n override var ongotpointercapture: ((PointerEvent) -> dynamic)?\n override var onlostpointercapture: ((PointerEvent) -> dynamic)?\n override var onpointerdown: ((PointerEvent) -> dynamic)?\n override var onpointermove: ((PointerEvent) -> dynamic)?\n override var onpointerup: ((PointerEvent) -> dynamic)?\n override var onpointercancel: ((PointerEvent) -> dynamic)?\n override var onpointerover: ((PointerEvent) -> dynamic)?\n override var onpointerout: ((PointerEvent) -> dynamic)?\n override var onpointerenter: ((PointerEvent) -> dynamic)?\n override var onpointerleave: ((PointerEvent) -> dynamic)?\n override var onabort: ((Event) -> dynamic)?\n override var onblur: ((FocusEvent) -> dynamic)?\n override var oncancel: ((Event) -> dynamic)?\n override var oncanplay: ((Event) -> dynamic)?\n override var oncanplaythrough: ((Event) -> dynamic)?\n override var onchange: ((Event) -> dynamic)?\n override var onclick: ((MouseEvent) -> dynamic)?\n override var onclose: ((Event) -> dynamic)?\n override var oncontextmenu: ((MouseEvent) -> dynamic)?\n override var oncuechange: ((Event) -> dynamic)?\n override var ondblclick: ((MouseEvent) -> dynamic)?\n override var ondrag: ((DragEvent) -> dynamic)?\n override var ondragend: ((DragEvent) -> dynamic)?\n override var ondragenter: ((DragEvent) -> dynamic)?\n override var ondragexit: ((DragEvent) -> dynamic)?\n override var ondragleave: ((DragEvent) -> dynamic)?\n override var ondragover: ((DragEvent) -> dynamic)?\n override var ondragstart: ((DragEvent) -> dynamic)?\n override var ondrop: ((DragEvent) -> dynamic)?\n override var ondurationchange: ((Event) -

```

```

> dynamic)?\r\n override var onemptied: ((Event) -> dynamic)?\r\n override var onended: ((Event) ->
dynamic)?\r\n override var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?\r\n override var onfocus:
((FocusEvent) -> dynamic)?\r\n override var oninput: ((InputEvent) -> dynamic)?\r\n override var oninvalid:
((Event) -> dynamic)?\r\n override var onkeydown: ((KeyboardEvent) -> dynamic)?\r\n override var
onkeypress: ((KeyboardEvent) -> dynamic)?\r\n override var onkeyup: ((KeyboardEvent) -> dynamic)?\r\n
override var onload: ((Event) -> dynamic)?\r\n override var onloadeddata: ((Event) -> dynamic)?\r\n override
var onloadedmetadata: ((Event) -> dynamic)?\r\n override var onloadend: ((Event) -> dynamic)?\r\n override var
onloadstart: ((ProgressEvent) -> dynamic)?\r\n override var onmousedown: ((MouseEvent) -> dynamic)?\r\n
override var onmouseenter: ((MouseEvent) -> dynamic)?\r\n override var onmouseleave: ((MouseEvent) ->
dynamic)?\r\n override var onmousemove: ((MouseEvent) -> dynamic)?\r\n override var onmouseout:
((MouseEvent) -> dynamic)?\r\n override var onmouseover: ((MouseEvent) -> dynamic)?\r\n override var
onmouseup: ((MouseEvent) -> dynamic)?\r\n override var onwheel: ((WheelEvent) -> dynamic)?\r\n override
var onpause: ((Event) -> dynamic)?\r\n override var onplay: ((Event) -> dynamic)?\r\n override var onplaying:
((Event) -> dynamic)?\r\n override var onprogress: ((ProgressEvent) -> dynamic)?\r\n override var
onratechange: ((Event) -> dynamic)?\r\n override var onreset: ((Event) -> dynamic)?\r\n override var onresize:
((Event) -> dynamic)?\r\n override var onscroll: ((Event) -> dynamic)?\r\n override var onseeked: ((Event) ->
dynamic)?\r\n override var onseeking: ((Event) -> dynamic)?\r\n override var onselect: ((Event) ->
dynamic)?\r\n override var onshow: ((Event) -> dynamic)?\r\n override var onstalled: ((Event) -> dynamic)?\r\n
override var onsubmit: ((Event) -> dynamic)?\r\n override var onsuspend: ((Event) -> dynamic)?\r\n override var
ontimeupdate: ((Event) -> dynamic)?\r\n override var ontoggle: ((Event) -> dynamic)?\r\n override var
onvolumechange: ((Event) -> dynamic)?\r\n override var onwaiting: ((Event) -> dynamic)?\r\n override var
oncopy: ((ClipboardEvent) -> dynamic)?\r\n override var oncut: ((ClipboardEvent) -> dynamic)?\r\n override
var onpaste: ((ClipboardEvent) -> dynamic)?\r\n override var contentEditable: String\r\n override val
isContentEditable: Boolean\r\n override fun prepend(vararg nodes: dynamic): Unit\r\n override fun
append(vararg nodes: dynamic): Unit\r\n override fun querySelector(selectors: String): Element?\r\n override
fun querySelectorAll(selectors: String): NodeList\r\n override fun before(vararg nodes: dynamic): Unit\r\n
override fun after(vararg nodes: dynamic): Unit\r\n override fun replaceWith(vararg nodes: dynamic): Unit\r\n
override fun remove(): Unit\r\n override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */):
Array<DOMQuad>\r\n override fun convertQuadFromNode(quad: dynamic, from: dynamic, options:
ConvertCoordinateOptions /* = definedExternally */): DOMQuad\r\n override fun convertRectFromNode(rect:
DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\r\n
override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* =
definedExternally */): DOMPoint\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[HTMLTextAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTextAreaElement) to Kotlin\r\n
*\r\n\r\npublic external abstract class HTMLTextAreaElement : HTMLElement {\r\n open var autocomplete:
String\r\n open var autofocus: Boolean\r\n open var cols: Int\r\n open var dirName: String\r\n open var
disabled: Boolean\r\n open val form: HTMLFormElement?\r\n open var inputMode: String\r\n open var
maxLength: Int\r\n open var minLength: Int\r\n open var name: String\r\n open var placeholder: String\r\n
open var readOnly: Boolean\r\n open var required: Boolean\r\n open var rows: Int\r\n open var wrap: String\r\n
open val type: String\r\n open var defaultValue: String\r\n open var value: String\r\n open val maxLength:
Int\r\n open val willValidate: Boolean\r\n open val validity: ValidityState\r\n open val validationMessage:
String\r\n open val labels: NodeList\r\n open var selectionStart: Int?\r\n open var selectionEnd: Int?\r\n open
var selectionDirection: String?\r\n fun checkValidity(): Boolean\r\n fun reportValidity(): Boolean\r\n fun
setCustomValidity(error: String): Unit\r\n fun select(): Unit\r\n fun setRangeText(replacement: String): Unit\r\n
fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally):
Unit\r\n fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally): Unit\r\n}\r\n\r\n/**
Exposes the JavaScript
[HTMLKeygenElement](https://developer.mozilla.org/en/docs/Web/API/HTMLKeygenElement) to Kotlin\r\n

```

```

*^public external abstract class HTMLKeygenElement : HTMLInputElement {
 open var autofocus: Boolean
 open var challenge: String
 open var disabled: Boolean
 open val form: HTMLFormElement?
 open var keytype: String
 open var name: String
 open val type: String
 open val willValidate: Boolean
 open val validity: ValidityState
 open val validationMessage: String
 open val labels: NodeList
 fun checkValidity(): Boolean
 fun reportValidity(): Boolean
 fun setCustomValidity(error: String):
 Unit
}

[HTMLOutputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOutputElement) to Kotlin

*^public external abstract class HTMLOutputElement : HTMLInputElement {
 open val htmlFor:
 DOMTokenList
 open val form: HTMLFormElement?
 open var name: String
 open val type:
 String
 open var defaultValue: String
 open var value: String
 open val willValidate: Boolean
 open val validity: ValidityState
 open val validationMessage: String
 open val labels: NodeList
 fun checkValidity(): Boolean
 fun reportValidity(): Boolean
 fun setCustomValidity(error: String):
 Unit
}

[HTMLProgressElement](https://developer.mozilla.org/en/docs/Web/API/HTMLProgressElement) to Kotlin

*^public external abstract class HTMLProgressElement : HTMLInputElement {
 open var value: Double
 open var max: Double
 open val position: Double
 open val labels: NodeList
}

[HTMLMeterElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMeterElement) to
Kotlin

*^public external abstract class HTMLMeterElement : HTMLInputElement {
 open var value:
 Double
 open var min: Double
 open var max: Double
 open var low: Double
 open var high:
 Double
 open var optimum: Double
 open val labels: NodeList
}

[HTMLFieldSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFieldSetElement) to Kotlin

*^public external abstract class HTMLFieldSetElement : HTMLInputElement {
 open var disabled: Boolean
 open val form: HTMLFormElement?
 open var name: String
 open val type: String
 open val
 elements: HTMLCollection
 open val willValidate: Boolean
 open val validity: ValidityState
 open val
 validationMessage: String
 fun checkValidity(): Boolean
 fun reportValidity(): Boolean
 fun
 setCustomValidity(error: String): Unit
}

[HTMLLegendElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLegendElement) to Kotlin

*^public external abstract class HTMLLegendElement : HTMLInputElement {
 open val form:
 HTMLFormElement?
 open var align: String
}

[ValidityState](https://developer.mozilla.org/en/docs/Web/API/ValidityState) to Kotlin

*^public external
abstract class ValidityState {
 open val valueMissing: Boolean
 open val typeMismatch: Boolean
 open val
 patternMismatch: Boolean
 open val tooLong: Boolean
 open val tooShort: Boolean
 open val
 rangeUnderflow: Boolean
 open val rangeOverflow: Boolean
 open val stepMismatch: Boolean
 open
 val badInput: Boolean
 open val customError: Boolean
 open val valid: Boolean
}

[HTMLDetailsElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDetailsElement) to Kotlin

*^public external abstract class HTMLDetailsElement : HTMLInputElement {
 open var open:
 Boolean
}

*^public external abstract class HTMLMenuElement : HTMLInputElement {
 open var type:
 String
 open var label: String
 open var compact: Boolean
}

*^public external abstract class
HTMLMenuItemElement : HTMLInputElement {
 open var type: String
 open var label: String
 open var
 icon: String
 open var disabled: Boolean
 open var checked: Boolean
 open var radiogroup: String
 open
 var default: Boolean
}

*^public external open class RelatedEvent(type: String, eventInitDict:
RelatedEventInit = definedExternally) : Event {
 open val relatedTarget: EventTarget?
}

*^public
external interface RelatedEventInit : EventInit {
 var relatedTarget: EventTarget? = null
}

*^
get() =
definedExternally
set(value) = definedExternally
}

@kotlin.internal.InlineOnly
*^public inline fun
RelatedEventInit(relatedTarget: EventTarget? = null, bubbles: Boolean? = false, cancelable: Boolean? = false,
composed: Boolean? = false): RelatedEventInit {
 val o = js("{}")
 o["relatedTarget"] =
 relatedTarget
 o["bubbles"] = bubbles
 o["cancelable"] = cancelable
 o["composed"] =
}

```





[TextMetrics](https://developer.mozilla.org/en/docs/Web/API/TextMetrics) to Kotlin

```

public external abstract class TextMetrics {
 open val width: Double
 open val actualBoundingBoxLeft: Double
 open val actualBoundingBoxRight: Double
 open val fontBoundingBoxAscent: Double
 open val fontBoundingBoxDescent: Double
 open val actualBoundingBoxAscent: Double
 open val actualBoundingBoxDescent: Double
 open val emHeightAscent: Double
 open val emHeightDescent: Double
 open val hangingBaseline: Double
 open val alphabeticBaseline: Double
 open val ideographicBaseline: Double
}

public external interface HitRegionOptions {
 var path: Path2D? /* = null */
 get() = definedExternally
 set(value) = definedExternally
 var fillRule: CanvasFillRule? /* = CanvasFillRule.NONZERO */
 get() = definedExternally
 set(value) = definedExternally
 var id: String? /* = "" */
 get() = definedExternally
 set(value) = definedExternally
 var parentID: String? /* = null */
 get() = definedExternally
 set(value) = definedExternally
 var cursor: String? /* = "inherit" */
 get() = definedExternally
 set(value) = definedExternally
 var control: Element? /* = null */
 get() = definedExternally
 set(value) = definedExternally
 var label: String? /* = null */
 get() = definedExternally
 set(value) = definedExternally
 var role: String? /* = null */
 get() = definedExternally
 set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun HitRegionOptions(path: Path2D? = null, fillRule: CanvasFillRule? = CanvasFillRule.NONZERO, id: String? = "", parentID: String? = null, cursor: String? = "inherit", control: Element? = null, label: String? = null, role: String? = null): HitRegionOptions {
 val o = js("{}")
 o["path"] = path
 o["fillRule"] = fillRule
 o["id"] = id
 o["parentID"] = parentID
 o["cursor"] = cursor
 o["control"] = control
 o["label"] = label
 o["role"] = role
 return o
}

* Exposes the JavaScript

```

[ImageData](https://developer.mozilla.org/en/docs/Web/API/ImageData) to Kotlin

```

public external open class ImageData : TexImageSource {
 constructor(sw: Int, sh: Int)
 constructor(data: Uint8ClampedArray, sw: Int, sh: Int = definedExternally)
 open val width: Int
 open val height: Int
 open val data: Uint8ClampedArray
}

* Exposes the JavaScript

```

[Path2D](https://developer.mozilla.org/en/docs/Web/API/Path2D) to Kotlin

```

public external open class Path2D() : CanvasPath {
 constructor(path: Path2D)
 constructor(paths: Array<Path2D>, fillRule: CanvasFillRule = definedExternally)
 constructor(d: String)
 fun addPath(path: Path2D, transform: dynamic = definedExternally): Unit
 override fun closePath(): Unit
 override fun moveTo(x: Double, y: Double): Unit
 override fun.lineTo(x: Double, y: Double): Unit
 override fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double): Unit
 override fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double): Unit
 override fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double): Unit
 override fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double): Unit
 override fun rect(x: Double, y: Double, w: Double, h: Double): Unit
 override fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean /* = definedExternally */): Unit
 override fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean /* = definedExternally */): Unit
}

* Exposes the JavaScript

```

[Touch](https://developer.mozilla.org/en/docs/Web/API/Touch) to Kotlin

```

public external abstract class Touch {
 open val region: String?
}

* Exposes the JavaScript

```

[ImageBitmapRenderingContext](https://developer.mozilla.org/en/docs/Web/API/ImageBitmapRenderingContext) to Kotlin

```

public external abstract class ImageBitmapRenderingContext {
 open val canvas: HTMLCanvasElement
 fun transferFromImageBitmap(bitmap: ImageBitmap?): Unit
}

public external interface ImageBitmapRenderingContextSettings {
 var alpha: Boolean? /* = true */
 get() = definedExternally
 set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ImageBitmapRenderingContextSettings(alpha: Boolean? = true): ImageBitmapRenderingContextSettings {
 val o = js("{}")
 o["alpha"] = alpha
 return o
}

* Exposes the JavaScript

```

[CustomElementRegistry](https://developer.mozilla.org/en/docs/Web/API/CustomElementRegistry) to Kotlin

```

*/\r\npublic external abstract class CustomElementRegistry {\r\n fun define(name: String, constructor: () ->
dynamic, options: ElementDefinitionOptions = definedExternally): Unit\r\n fun get(name: String): Any?\r\n fun
whenDefined(name: String): Promise<Unit>\r\n}\r\n\r\npublic external interface ElementDefinitionOptions {\r\n
var extends: String?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ElementDefinitionOptions(extends:
String? = undefined): ElementDefinitionOptions {\r\n val o = js("{}")\r\n o["extends"] = extends\r\n\r\n
return o\r\n}\r\n\r\npublic external interface ElementContentEditable {\r\n var contentEditable: String\r\n val
isContentEditable: Boolean\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[DataTransfer](https://developer.mozilla.org/en/docs/Web/API/DataTransfer) to Kotlin\r\n*/\r\npublic external
abstract class DataTransfer {\r\n open var dropEffect: String\r\n open var effectAllowed: String\r\n open val
items: DataTransferItemList\r\n open val types: Array<out String>\r\n open val files: FileList\r\n fun
setDragImage(image: Element, x: Int, y: Int): Unit\r\n fun getData(format: String): String\r\n fun setData(format:
String, data: String): Unit\r\n fun clearData(format: String = definedExternally): Unit\r\n}\r\n\r\n/**\r\n * Exposes
the JavaScript [DataTransferItemList](https://developer.mozilla.org/en/docs/Web/API/DataTransferItemList) to
Kotlin\r\n*/\r\npublic external abstract class DataTransferItemList {\r\n open val length: Int\r\n fun add(data:
String, type: String): DataTransferItem?\r\n fun add(data: File): DataTransferItem?\r\n fun remove(index: Int):
Unit\r\n fun clear(): Unit\r\n}\r\n\r\n@kotlin.internal.InlineOnly inline operator fun DataTransferItemList.get(index:
Int): DataTransferItem? = asDynamic()[index]\r\n\r\n/**\r\n * Exposes the JavaScript
[DataTransferItem](https://developer.mozilla.org/en/docs/Web/API/DataTransferItem) to Kotlin\r\n*/\r\npublic
external abstract class DataTransferItem {\r\n open val kind: String\r\n open val type: String\r\n fun
getAsString(_callback: ((String) -> Unit)?): Unit\r\n fun getAsFile(): File?\r\n}\r\n\r\n/**\r\n * Exposes the
JavaScript [DragEvent](https://developer.mozilla.org/en/docs/Web/API/TouchEvent) to Kotlin\r\n*/\r\npublic
external open class DragEvent(type: String, eventInitDict: DragEventInit = definedExternally) : MouseEvent {\r\n
open val dataTransfer: DataTransfer?\r\n}\r\n\r\npublic external interface DragEventInit : MouseEventInit {\r\n
var dataTransfer: DataTransfer? /* = null */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun DragEventInit(dataTransfer:
DataTransfer? = null, screenX: Int? = 0, screenY: Int? = 0, clientX: Int? = 0, clientY: Int? = 0, button: Short? = 0,
buttons: Short? = 0, relatedTarget: EventTarget? = null, ctrlKey: Boolean? = false, shiftKey: Boolean? = false,
altKey: Boolean? = false, metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock:
Boolean? = false, modifierFn: Boolean? = false, modifierFnLock: Boolean? = false, modifierHyper: Boolean? =
false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false, modifierSuper: Boolean? = false,
modifierSymbol: Boolean? = false, modifierSymbolLock: Boolean? = false, view: Window? = null, detail: Int? = 0,
bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): DragEventInit {\r\n val o =
js("{}")\r\n o["dataTransfer"] = dataTransfer\r\n o["screenX"] = screenX\r\n o["screenY"] =
screenY\r\n o["clientX"] = clientX\r\n o["clientY"] = clientY\r\n o["button"] = button\r\n o["buttons"]
= buttons\r\n o["relatedTarget"] = relatedTarget\r\n o["ctrlKey"] = ctrlKey\r\n o["shiftKey"] = shiftKey\r\n
 o["altKey"] = altKey\r\n o["metaKey"] = metaKey\r\n o["modifierAltGraph"] = modifierAltGraph\r\n
 o["modifierCapsLock"] = modifierCapsLock\r\n o["modifierFn"] = modifierFn\r\n o["modifierFnLock"] =
modifierFnLock\r\n o["modifierHyper"] = modifierHyper\r\n o["modifierNumLock"] =
modifierNumLock\r\n o["modifierScrollLock"] = modifierScrollLock\r\n o["modifierSuper"] =
modifierSuper\r\n o["modifierSymbol"] = modifierSymbol\r\n o["modifierSymbolLock"] =
modifierSymbolLock\r\n o["view"] = view\r\n o["detail"] = detail\r\n o["bubbles"] = bubbles\r\n
 o["cancelable"] = cancelable\r\n o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\npublic external
abstract class BarProp {\r\n open val visible: Boolean\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[History](https://developer.mozilla.org/en/docs/Web/API/History) to Kotlin\r\n*/\r\npublic external abstract class
History {\r\n open val length: Int\r\n open var scrollRestoration: ScrollRestoration\r\n open val state: Any?\r\n
fun go(delta: Int = definedExternally): Unit\r\n fun back(): Unit\r\n fun forward(): Unit\r\n fun pushState(data:
Any?, title: String, url: String? = definedExternally): Unit\r\n fun replaceState(data: Any?, title: String, url: String?)

```

```

= definedExternally): Unit\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[Location](https://developer.mozilla.org/en/docs/Web/API/Location) to Kotlin\r\n *^\r\npublic external abstract
class Location {\r\n open var href: String\r\n open val origin: String\r\n open var protocol: String\r\n open
var host: String\r\n open var hostname: String\r\n open var port: String\r\n open var pathname: String\r\n
open var search: String\r\n open var hash: String\r\n open val ancestorOrigins: Array<out String>\r\n fun
assign(url: String): Unit\r\n fun replace(url: String): Unit\r\n fun reload(): Unit\r\n}\r\n\r\n/**\r\n * Exposes the
JavaScript [PopStateEvent](https://developer.mozilla.org/en/docs/Web/API/PopStateEvent) to Kotlin\r\n
*^\r\npublic external open class PopStateEvent(type: String, eventInitDict: PopStateEventInit = definedExternally) :
Event {\r\n open val state: Any?\r\n}\r\n\r\npublic external interface PopStateEventInit : EventInit {\r\n var state:
Any? /* = null */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun PopStateEventInit(state: Any? = null,
bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): PopStateEventInit {\r\n val
o = js("{}")\r\n\r\n o["state"] = state\r\n o["bubbles"] = bubbles\r\n o["cancelable"] = cancelable\r\n
o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[HashChangeEvent](https://developer.mozilla.org/en/docs/Web/API/HashChangeEvent) to Kotlin\r\n *^\r\npublic
external open class HashChangeEvent(type: String, eventInitDict: HashChangeEventInit = definedExternally) :
Event {\r\n open val oldURL: String\r\n open val newURL: String\r\n}\r\n\r\npublic external interface
HashChangeEventInit : EventInit {\r\n var oldURL: String? /* = "" */\r\n get() = definedExternally\r\n
set(value) = definedExternally\r\n var newURL: String? /* = "" */\r\n get() = definedExternally\r\n
set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
HashChangeEventInit(oldURL: String? = "", newURL: String? = "", bubbles: Boolean? = false, cancelable:
Boolean? = false, composed: Boolean? = false): HashChangeEventInit {\r\n val o = js("{}")\r\n\r\n
o["oldURL"] = oldURL\r\n o["newURL"] = newURL\r\n o["bubbles"] = bubbles\r\n o["cancelable"] =
cancelable\r\n o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[PageTransitionEvent](https://developer.mozilla.org/en/docs/Web/API/PageTransitionEvent) to Kotlin\r\n
*^\r\npublic external open class PageTransitionEvent(type: String, eventInitDict: PageTransitionEventInit =
definedExternally) : Event {\r\n open val persisted: Boolean\r\n}\r\n\r\npublic external interface
PageTransitionEventInit : EventInit {\r\n var persisted: Boolean? /* = false */\r\n get() = definedExternally\r\n
set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
PageTransitionEventInit(persisted: Boolean? = false, bubbles: Boolean? = false, cancelable: Boolean? = false,
composed: Boolean? = false): PageTransitionEventInit {\r\n val o = js("{}")\r\n\r\n o["persisted"] =
persisted\r\n o["bubbles"] = bubbles\r\n o["cancelable"] = cancelable\r\n o["composed"] =
composed\r\n\r\n return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[BeforeUnloadEvent](https://developer.mozilla.org/en/docs/Web/API/BeforeUnloadEvent) to Kotlin\r\n *^\r\npublic
external open class BeforeUnloadEvent : Event {\r\n var returnValue: String\r\n}\r\n\r\npublic external abstract
class ApplicationCache : EventTarget {\r\n open val status: Short\r\n open var onchecking: ((Event) ->
dynamic)?\r\n open var onerror: ((Event) -> dynamic)?\r\n open var onnoupdate: ((Event) -> dynamic)?\r\n
open var ondownloading: ((Event) -> dynamic)?\r\n open var onprogress: ((ProgressEvent) -> dynamic)?\r\n
open var onupdateready: ((Event) -> dynamic)?\r\n open var oncached: ((Event) -> dynamic)?\r\n open var
onobsolete: ((Event) -> dynamic)?\r\n fun update(): Unit\r\n fun abort(): Unit\r\n fun swapCache(): Unit\r\n\r\n
companion object {\r\n val UNCACHED: Short\r\n val IDLE: Short\r\n val CHECKING: Short\r\n
val DOWNLOADING: Short\r\n val UPDATEREADY: Short\r\n val OBSOLETE: Short\r\n
}\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[NavigatorOnLine](https://developer.mozilla.org/en/docs/Web/API/NavigatorOnLine) to Kotlin\r\n *^\r\npublic
external interface NavigatorOnLine {\r\n val onLine: Boolean\r\n}\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[ErrorEvent](https://developer.mozilla.org/en/docs/Web/API/ErrorEvent) to Kotlin\r\n *^\r\npublic external open
class ErrorEvent(type: String, eventInitDict: ErrorEventInit = definedExternally) : Event {\r\n open val message:
String\r\n open val filename: String\r\n open val lineno: Int\r\n open val colno: Int\r\n open val error:

```



```

Any? \r\n} \r\n\r\npublic external interface ErrorEventInit : EventInit { \r\n var message: String? /* = "" */ \r\n
get() = definedExternally \r\n set(value) = definedExternally \r\n var filename: String? /* = "" */ \r\n get()
= definedExternally \r\n set(value) = definedExternally \r\n var lineno: Int? /* = 0 */ \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n var colno: Int? /* = 0 */ \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n var error: Any? /* = null */ \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n} \r\n\r\n@kotlin.internal.InlineOnly \r\n\r\npublic inline fun
ErrorEventInit(message: String? = "", filename: String? = "", lineno: Int? = 0, colno: Int? = 0, error: Any? = null,
bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): ErrorEventInit { \r\n val o =
js("{}") \r\n\r\n o["message"] = message \r\n o["filename"] = filename \r\n o["lineno"] = lineno \r\n
o["colno"] = colno \r\n o["error"] = error \r\n o["bubbles"] = bubbles \r\n o["cancelable"] = cancelable \r\n
o["composed"] = composed \r\n\r\n return o \r\n} \r\n\r\n\r\n/* Exposes the JavaScript
[PromiseRejectionEvent](https://developer.mozilla.org/en/docs/Web/API/PromiseRejectionEvent) to Kotlin \r\n
*/ \r\n\r\npublic external open class PromiseRejectionEvent(type: String, eventInitDict: PromiseRejectionEventInit) :
Event { \r\n open val promise: Promise<Any?> \r\n open val reason: Any? \r\n\r\n\r\npublic external interface
PromiseRejectionEventInit : EventInit { \r\n var promise: Promise<Any?>? \r\n get() = definedExternally \r\n
set(value) = definedExternally \r\n var reason: Any? \r\n get() = definedExternally \r\n set(value) =
definedExternally \r\n} \r\n\r\n\r\n@kotlin.internal.InlineOnly \r\n\r\npublic inline fun PromiseRejectionEventInit(promise:
Promise<Any?>?, reason: Any? = undefined, bubbles: Boolean? = false, cancelable: Boolean? = false, composed:
Boolean? = false): PromiseRejectionEventInit { \r\n val o = js("{}") \r\n\r\n o["promise"] = promise \r\n
o["reason"] = reason \r\n o["bubbles"] = bubbles \r\n o["cancelable"] = cancelable \r\n o["composed"] =
composed \r\n\r\n return o \r\n} \r\n\r\n\r\n/* Exposes the JavaScript
[GlobalEventHandlers](https://developer.mozilla.org/en/docs/Web/API/GlobalEventHandlers) to Kotlin \r\n
*/ \r\n\r\npublic external interface GlobalEventHandlers { \r\n var ongotpointercapture: ((PointerEvent) ->
dynamic)? \r\n var onlostpointercapture: ((PointerEvent) -> dynamic)? \r\n var onpointerdown: ((PointerEvent) ->
dynamic)? \r\n var onpointermove: ((PointerEvent) -> dynamic)? \r\n var onpointerup: ((PointerEvent) ->
dynamic)? \r\n var onpointercancel: ((PointerEvent) -> dynamic)? \r\n var onpointerover: ((PointerEvent) ->
dynamic)? \r\n var onpointerout: ((PointerEvent) -> dynamic)? \r\n var onpointerenter: ((PointerEvent) ->
dynamic)? \r\n var onpointerleave: ((PointerEvent) -> dynamic)? \r\n var onabort: ((Event) -> dynamic)? \r\n var
onblur: ((FocusEvent) -> dynamic)? \r\n var oncancel: ((Event) -> dynamic)? \r\n var oncanplay: ((Event) ->
dynamic)? \r\n var oncanplaythrough: ((Event) -> dynamic)? \r\n var onchange: ((Event) -> dynamic)? \r\n var
onclick: ((MouseEvent) -> dynamic)? \r\n var onclose: ((Event) -> dynamic)? \r\n var oncontextmenu:
((MouseEvent) -> dynamic)? \r\n var oncuechange: ((Event) -> dynamic)? \r\n var ondblclick: ((MouseEvent) ->
dynamic)? \r\n var ondrag: ((DragEvent) -> dynamic)? \r\n var ondragend: ((DragEvent) -> dynamic)? \r\n var
ondragenter: ((DragEvent) -> dynamic)? \r\n var ondragexit: ((DragEvent) -> dynamic)? \r\n var ondragleave:
((DragEvent) -> dynamic)? \r\n var ondragover: ((DragEvent) -> dynamic)? \r\n var ondragstart: ((DragEvent) ->
dynamic)? \r\n var ondrop: ((DragEvent) -> dynamic)? \r\n var ondurationchange: ((Event) -> dynamic)? \r\n var
onemptied: ((Event) -> dynamic)? \r\n var onended: ((Event) -> dynamic)? \r\n var onerror: ((dynamic, String, Int,
Int, Any?) -> dynamic)? \r\n var onfocus: ((FocusEvent) -> dynamic)? \r\n var oninput: ((InputEvent) ->
dynamic)? \r\n var oninvalid: ((Event) -> dynamic)? \r\n var onkeydown: ((KeyboardEvent) -> dynamic)? \r\n
var onkeypress: ((KeyboardEvent) -> dynamic)? \r\n var onkeyup: ((KeyboardEvent) -> dynamic)? \r\n var
onload: ((Event) -> dynamic)? \r\n var onloadeddata: ((Event) -> dynamic)? \r\n var onloadedmetadata: ((Event) -
> dynamic)? \r\n var onloadend: ((Event) -> dynamic)? \r\n var onloadstart: ((ProgressEvent) -> dynamic)? \r\n
var onmousedown: ((MouseEvent) -> dynamic)? \r\n var onmouseenter: ((MouseEvent) -> dynamic)? \r\n var
onmouseleave: ((MouseEvent) -> dynamic)? \r\n var onmousemove: ((MouseEvent) -> dynamic)? \r\n var
onmouseout: ((MouseEvent) -> dynamic)? \r\n var onmouseover: ((MouseEvent) -> dynamic)? \r\n var
onmouseup: ((MouseEvent) -> dynamic)? \r\n var onwheel: ((WheelEvent) -> dynamic)? \r\n var onpause:
((Event) -> dynamic)? \r\n var onplay: ((Event) -> dynamic)? \r\n var onplaying: ((Event) -> dynamic)? \r\n var
onprogress: ((ProgressEvent) -> dynamic)? \r\n var onratechange: ((Event) -> dynamic)? \r\n var onreset: ((Event)

```

-> dynamic)?\r\n var onresize: ((Event) -> dynamic)?\r\n var onscroll: ((Event) -> dynamic)?\r\n var onseeked: ((Event) -> dynamic)?\r\n var onseeking: ((Event) -> dynamic)?\r\n var onselect: ((Event) -> dynamic)?\r\n var onshow: ((Event) -> dynamic)?\r\n var onstalled: ((Event) -> dynamic)?\r\n var onsubmit: ((Event) -> dynamic)?\r\n var onsuspend: ((Event) -> dynamic)?\r\n var ontimeupdate: ((Event) -> dynamic)?\r\n var ontoggle: ((Event) -> dynamic)?\r\n var onvolumechange: ((Event) -> dynamic)?\r\n var onwaiting: ((Event) -> dynamic)?\r\n}\r\n\r\n/\*\*\r\n \* Exposes the JavaScript

[WindowEventHandlers](https://developer.mozilla.org/en/docs/Web/API/WindowEventHandlers) to Kotlin\r\n\r\n\*/\r\n\r\npublic external interface WindowEventHandlers {\r\n var onafterprint: ((Event) -> dynamic)?\r\n var onbeforeprint: ((Event) -> dynamic)?\r\n var onbeforeunload: ((BeforeUnloadEvent) -> String?)?\r\n var onhashchange: ((HashChangeEvent) -> dynamic)?\r\n var onlanguagechange: ((Event) -> dynamic)?\r\n var onmessage: ((MessageEvent) -> dynamic)?\r\n var onoffline: ((Event) -> dynamic)?\r\n var online: ((Event) -> dynamic)?\r\n var onpagehide: ((PageTransitionEvent) -> dynamic)?\r\n var onpageshow: ((PageTransitionEvent) -> dynamic)?\r\n var onpopstate: ((PopStateEvent) -> dynamic)?\r\n var onrejectionhandled: ((Event) -> dynamic)?\r\n var onstorage: ((StorageEvent) -> dynamic)?\r\n var onunhandledrejection: ((PromiseRejectionEvent) -> dynamic)?\r\n var onunload: ((Event) -> dynamic)?\r\n}\r\n\r\npublic external interface DocumentAndElementEventHandlers {\r\n var oncopy: ((ClipboardEvent) -> dynamic)?\r\n var oncut: ((ClipboardEvent) -> dynamic)?\r\n var onpaste: ((ClipboardEvent) -> dynamic)?\r\n}\r\n\r\n/\*\*\r\n \* Exposes the JavaScript

[WindowOrWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/WindowOrWorkerGlobalScope) to Kotlin\r\n\r\n\*/\r\n\r\npublic external interface WindowOrWorkerGlobalScope {\r\n val caches: CacheStorage\r\n val origin: String\r\n fun fetch(input: dynamic, init: RequestInit = definedExternally): Promise<Response>\r\n fun btoa(data: String): String\r\n fun atob(data: String): String\r\n fun setTimeout(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int\r\n fun clearTimeout(handle: Int = definedExternally): Unit\r\n fun setInterval(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int\r\n fun clearInterval(handle: Int = definedExternally): Unit\r\n fun createImageBitmap(image: dynamic, options: ImageBitmapOptions = definedExternally): Promise<ImageBitmap>\r\n fun createImageBitmap(image: dynamic, sx: Int, sy: Int, sw: Int, sh: Int, options: ImageBitmapOptions = definedExternally): Promise<ImageBitmap>\r\n}\r\n\r\n/\*\*\r\n \* Exposes the JavaScript

[NavigatorID](https://developer.mozilla.org/en/docs/Web/API/NavigatorID) to Kotlin\r\n\r\n\*/\r\n\r\npublic external interface NavigatorID {\r\n val appCodeName: String\r\n val appName: String\r\n val appVersion: String\r\n val platform: String\r\n val product: String\r\n val productSub: String\r\n val userAgent: String\r\n val vendor: String\r\n val vendorSub: String\r\n val oscpu: String\r\n fun taintEnabled(): Boolean\r\n}\r\n\r\n/\*\*\r\n \* Exposes the JavaScript

[NavigatorLanguage](https://developer.mozilla.org/en/docs/Web/API/NavigatorLanguage) to Kotlin\r\n\r\n\*/\r\n\r\npublic external interface NavigatorLanguage {\r\n val language: String\r\n val languages: Array<out String>\r\n}\r\n\r\npublic external interface NavigatorContentUtils {\r\n fun registerProtocolHandler(scheme: String, url: String, title: String): Unit\r\n fun registerContentHandler(mimeType: String, url: String, title: String): Unit\r\n fun isProtocolHandlerRegistered(scheme: String, url: String): String\r\n fun isContentHandlerRegistered(mimeType: String, url: String): String\r\n fun unregisterProtocolHandler(scheme: String, url: String): Unit\r\n fun unregisterContentHandler(mimeType: String, url: String): Unit\r\n}\r\n\r\npublic external interface NavigatorCookies {\r\n val cookieEnabled: Boolean\r\n}\r\n\r\n/\*\*\r\n \* Exposes the JavaScript

[NavigatorPlugins](https://developer.mozilla.org/en/docs/Web/API/NavigatorPlugins) to Kotlin\r\n\r\n\*/\r\n\r\npublic external interface NavigatorPlugins {\r\n val plugins: PluginArray\r\n val mimeTypes: MimeTypes\r\n fun javaEnabled(): Boolean\r\n}\r\n\r\n/\*\*\r\n \* Exposes the JavaScript

[PluginArray](https://developer.mozilla.org/en/docs/Web/API/PluginArray) to Kotlin\r\n\r\n\*/\r\n\r\npublic external abstract class PluginArray : ItemArrayLike<Plugin> {\r\n override val length: Int\r\n fun refresh(reload: Boolean = definedExternally): Unit\r\n override fun item(index: Int): Plugin?\r\n fun namedItem(name: String): Plugin?\r\n}\r\n\r\n@kotlin.internal.InlineOnly inline operator fun PluginArray.get(index: Int): Plugin? =



UnionMessagePortOrWindow? = null, ports: Array<MessagePort>? = arrayOf(), bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): MessageEventInit {\r\n val o = js("{}")\r\n o["data"] = data\r\n o["origin"] = origin\r\n o["lastEventId"] = lastEventId\r\n o["source"] = source\r\n o["ports"] = ports\r\n o["bubbles"] = bubbles\r\n o["cancelable"] = cancelable\r\n o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript

[EventSource](<https://developer.mozilla.org/en/docs/Web/API/EventSource>) to Kotlin\r\n \* \r\n\r\npublic external open class EventSource(url: String, eventSourceInitDict: EventSourceInit = definedExternally) : EventTarget {\r\n open val url: String\r\n open val withCredentials: Boolean\r\n open val readyState: Short\r\n var onopen: ((Event) -> dynamic)?\r\n var onmessage: ((MessageEvent) -> dynamic)?\r\n var onerror: ((Event) -> dynamic)?\r\n fun close(): Unit\r\n\r\n companion object {\r\n val CONNECTING: Short\r\n val OPEN: Short\r\n val CLOSED: Short\r\n }\r\n}\r\n\r\n\r\npublic external interface EventSourceInit {\r\n var withCredentials: Boolean? /\* = false \*/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun EventSourceInit(withCredentials: Boolean? = false): EventSourceInit {\r\n val o = js("{}")\r\n o["withCredentials"] = withCredentials\r\n\r\n return o\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript

[WebSocket](<https://developer.mozilla.org/en/docs/Web/API/WebSocket>) to Kotlin\r\n \* \r\n\r\npublic external open class WebSocket(url: String, protocols: dynamic = definedExternally) : EventTarget {\r\n open val url: String\r\n open val readyState: Short\r\n open val bufferedAmount: Int\r\n var onopen: ((Event) -> dynamic)?\r\n var onerror: ((Event) -> dynamic)?\r\n var onclose: ((Event) -> dynamic)?\r\n open val extensions: String\r\n open val protocol: String\r\n var onmessage: ((MessageEvent) -> dynamic)?\r\n var binaryType: BinaryType\r\n fun close(code: Short = definedExternally, reason: String = definedExternally): Unit\r\n fun send(data: String): Unit\r\n fun send(data: Blob): Unit\r\n fun send(data: ArrayBuffer): Unit\r\n fun send(data: ArrayBufferView): Unit\r\n\r\n\r\n companion object {\r\n val CONNECTING: Short\r\n val OPEN: Short\r\n val CLOSING: Short\r\n val CLOSED: Short\r\n }\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript

[CloseEvent](<https://developer.mozilla.org/en/docs/Web/API/CloseEvent>) to Kotlin\r\n \* \r\n\r\npublic external open class CloseEvent(type: String, eventInitDict: CloseEventInit = definedExternally) : Event {\r\n open val wasClean: Boolean\r\n open val code: Short\r\n open val reason: String\r\n}\r\n\r\n\r\npublic external interface CloseEventInit : EventInit {\r\n var wasClean: Boolean? /\* = false \*/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var code: Short? /\* = 0 \*/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var reason: String? /\* = "" \*/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\npublic inline fun CloseEventInit(wasClean: Boolean? = false, code: Short? = 0, reason: String? = "", bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): CloseEventInit {\r\n val o = js("{}")\r\n o["wasClean"] = wasClean\r\n o["code"] = code\r\n o["reason"] = reason\r\n o["bubbles"] = bubbles\r\n o["cancelable"] = cancelable\r\n o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript

[MessageChannel](<https://developer.mozilla.org/en/docs/Web/API/MessageChannel>) to Kotlin\r\n \* \r\n\r\npublic external open class MessageChannel {\r\n open val port1: MessagePort\r\n open val port2: MessagePort\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript

[MessagePort](<https://developer.mozilla.org/en/docs/Web/API/MessagePort>) to Kotlin\r\n \* \r\n\r\npublic external abstract class MessagePort : EventTarget, UnionMessagePortOrWindow, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {\r\n open var onmessage: ((MessageEvent) -> dynamic)?\r\n fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally): Unit\r\n fun start(): Unit\r\n fun close(): Unit\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript

[BroadcastChannel](<https://developer.mozilla.org/en/docs/Web/API/BroadcastChannel>) to Kotlin\r\n \* \r\n\r\npublic external open class BroadcastChannel(name: String) : EventTarget {\r\n open val name: String\r\n var onmessage: ((MessageEvent) -> dynamic)?\r\n fun postMessage(message: Any?): Unit\r\n fun close(): Unit\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript

[WorkerGlobalScope](<https://developer.mozilla.org/en/docs/Web/API/WorkerGlobalScope>) to Kotlin\r\n

```

*/\r\npublic external abstract class WorkerGlobalScope : EventTarget, WindowOrWorkerGlobalScope,
GlobalPerformance {\r\n open val self: WorkerGlobalScope\r\n open val location: WorkerLocation\r\n open
val navigator: WorkerNavigator\r\n open var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?\r\n open
var onlanguagechange: ((Event) -> dynamic)?\r\n open var onoffline: ((Event) -> dynamic)?\r\n open var
ononline: ((Event) -> dynamic)?\r\n open var onrejectionhandled: ((Event) -> dynamic)?\r\n open var
onunhandledrejection: ((PromiseRejectionEvent) -> dynamic)?\r\n fun importScripts(vararg urls: String):
Unit\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[DedicatedWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/DedicatedWorkerGlobalScope) to
Kotlin\r\n */\r\npublic external abstract class DedicatedWorkerGlobalScope : WorkerGlobalScope {\r\n open var
onmessage: ((MessageEvent) -> dynamic)?\r\n fun postMessage(message: Any?, transfer: Array<dynamic> =
definedExternally): Unit\r\n fun close(): Unit\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[SharedWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/SharedWorkerGlobalScope) to
Kotlin\r\n */\r\npublic external abstract class SharedWorkerGlobalScope : WorkerGlobalScope {\r\n open val
name: String\r\n open val applicationCache: ApplicationCache\r\n open var onconnect: ((Event) ->
dynamic)?\r\n fun close(): Unit\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[AbstractWorker](https://developer.mozilla.org/en/docs/Web/API/AbstractWorker) to Kotlin\r\n */\r\npublic
external interface AbstractWorker {\r\n var onerror: ((Event) -> dynamic)?\r\n}\r\n\r\n/**\r\n * Exposes the
JavaScript [Worker](https://developer.mozilla.org/en/docs/Web/API/Worker) to Kotlin\r\n */\r\npublic external
open class Worker(scriptURL: String, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker
{\r\n var onmessage: ((MessageEvent) -> dynamic)?\r\n override var onerror: ((Event) -> dynamic)?\r\n fun
terminate(): Unit\r\n fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally):
Unit\r\n}\r\n\r\npublic external interface WorkerOptions {\r\n var type: WorkerType? /* = WorkerType.CLASSIC
*/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var credentials:
RequestCredentials? /* = RequestCredentials.OMIT */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun WorkerOptions(type: WorkerType? =
WorkerType.CLASSIC, credentials: RequestCredentials? = RequestCredentials.OMIT): WorkerOptions {\r\n val
o = js("{}")\r\n o["type"] = type\r\n o["credentials"] = credentials\r\n return o\r\n}\r\n\r\n/**\r\n *
Exposes the JavaScript [SharedWorker](https://developer.mozilla.org/en/docs/Web/API/SharedWorker) to
Kotlin\r\n */\r\npublic external open class SharedWorker(scriptURL: String, name: String = definedExternally,
options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {\r\n open val port: MessagePort\r\n
override var onerror: ((Event) -> dynamic)?\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[NavigatorConcurrentHardware](https://developer.mozilla.org/en/docs/Web/API/NavigatorConcurrentHardware) to
Kotlin\r\n */\r\npublic external interface NavigatorConcurrentHardware {\r\n val hardwareConcurrency:
Int\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[WorkerNavigator](https://developer.mozilla.org/en/docs/Web/API/WorkerNavigator) to Kotlin\r\n */\r\npublic
external abstract class WorkerNavigator : NavigatorID, NavigatorLanguage, NavigatorOnLine,
NavigatorConcurrentHardware {\r\n open val serviceWorker: ServiceWorkerContainer\r\n}\r\n\r\n/**\r\n *
Exposes the JavaScript [WorkerLocation](https://developer.mozilla.org/en/docs/Web/API/WorkerLocation) to
Kotlin\r\n */\r\npublic external abstract class WorkerLocation {\r\n open var href: String\r\n open val origin:
String\r\n open val protocol: String\r\n open val host: String\r\n open val hostname: String\r\n open val port:
String\r\n open val pathname: String\r\n open val search: String\r\n open val hash: String\r\n}\r\n\r\n/**\r\n *
Exposes the JavaScript [Storage](https://developer.mozilla.org/en/docs/Web/API/Storage) to Kotlin\r\n */\r\npublic
external abstract class Storage {\r\n open val length: Int\r\n fun key(index: Int): String?\r\n fun getItem(key:
String): String?\r\n fun setItem(key: String, value: String): Unit\r\n fun removeItem(key: String): Unit\r\n fun
clear(): Unit\r\n}\r\n\r\n@kotlin.internal.InlineOnly inline operator fun Storage.get(key: String): String? =
asDynamic()[key]\r\n\r\n@kotlin.internal.InlineOnly inline operator fun Storage.set(key: String, value: String): Unit {
asDynamic()[key] = value; }\r\n\r\n/**\r\n * Exposes the JavaScript
[WindowSessionStorage](https://developer.mozilla.org/en/docs/Web/API/WindowSessionStorage) to Kotlin\r\n

```

```

*/\r\npublic external interface WindowSessionStorage {\r\n val sessionStorage: Storage\r\n}\r\n\r\n**\r\n\r\n * Exposes the JavaScript
[WindowLocalStorage](https://developer.mozilla.org/en/docs/Web/API/WindowLocalStorage) to Kotlin\r\n
*/\r\npublic external interface WindowLocalStorage {\r\n val localStorage: Storage\r\n}\r\n\r\n**\r\n\r\n * Exposes
the JavaScript [StorageEvent](https://developer.mozilla.org/en/docs/Web/API/StorageEvent) to Kotlin\r\n
*/\r\npublic external open class StorageEvent(type: String, eventInitDict: StorageEventInit = definedExternally) :
Event {\r\n open val key: String?\r\n open val oldValue: String?\r\n open val newValue: String?\r\n open val
url: String\r\n open val storageArea: Storage?\r\n}\r\n\r\npublic external interface StorageEventInit : EventInit
{\r\n var key: String? /* = null */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var
oldValue: String? /* = null */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var
newValue: String? /* = null */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var url:
String? /* = "" */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var storageArea:
Storage? /* = null */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun StorageEventInit(key: String? = null,
oldValue: String? = null, newValue: String? = null, url: String? = "", storageArea: Storage? = null, bubbles:
Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): StorageEventInit {\r\n val o =
js("{}")\r\n\r\n o["key"] = key\r\n o["oldValue"] = oldValue\r\n o["newValue"] = newValue\r\n o["url"] = url\r\n o["storageArea"] = storageArea\r\n o["bubbles"] = bubbles\r\n o["cancelable"] =
cancelable\r\n o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\npublic external abstract class
HTMLAppletElement : HTMLElement {\r\n open var align: String\r\n open var alt: String\r\n open var
archive: String\r\n open var code: String\r\n open var codeBase: String\r\n open var height: String\r\n open
var hspace: Int\r\n open var name: String\r\n open var _object: String\r\n open var vspace: Int\r\n open var
width: String\r\n}\r\n\r\n**\r\n\r\n * Exposes the JavaScript
[HTMLMarqueeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMarqueeElement) to Kotlin\r\n
*/\r\npublic external abstract class HTMLMarqueeElement : HTMLElement {\r\n open var behavior: String\r\n
open var bgColor: String\r\n open var direction: String\r\n open var height: String\r\n open var hspace: Int\r\n
open var loop: Int\r\n open var scrollAmount: Int\r\n open var scrollDelay: Int\r\n open var trueSpeed:
Boolean\r\n open var vspace: Int\r\n open var width: String\r\n open var onbounce: ((Event) -> dynamic)?\r\n
open var onfinish: ((Event) -> dynamic)?\r\n open var onstart: ((Event) -> dynamic)?\r\n fun start(): Unit\r\n
fun stop(): Unit\r\n}\r\n\r\n**\r\n\r\n * Exposes the JavaScript
[HTMLFrameSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFrameSetElement) to Kotlin\r\n
*/\r\npublic external abstract class HTMLFrameSetElement : HTMLElement, WindowEventHandlers {\r\n open
var cols: String\r\n open var rows: String\r\n}\r\n\r\npublic external abstract class HTMLFrameElement :
HTMLElement {\r\n open var name: String\r\n open var scrolling: String\r\n open var src: String\r\n open
var frameBorder: String\r\n open var longDesc: String\r\n open var noResize: Boolean\r\n open val
contentDocument: Document?\r\n open val contentWindow: Window?\r\n open var marginHeight: String\r\n
open var marginWidth: String\r\n}\r\n\r\npublic external abstract class HTMLDirectoryElement : HTMLElement
{\r\n open var compact: Boolean\r\n}\r\n\r\n**\r\n\r\n * Exposes the JavaScript
[HTMLFontElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFontElement) to Kotlin\r\n
*/\r\npublic
external abstract class HTMLFontElement : HTMLElement {\r\n open var color: String\r\n open var face:
String\r\n open var size: String\r\n}\r\n\r\npublic external interface External {\r\n fun AddSearchProvider():
Unit\r\n fun IsSearchProviderInstalled(): Unit\r\n}\r\n\r\npublic external interface EventInit {\r\n var bubbles:
Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var cancelable:
Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var composed:
Boolean? /* = false */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun EventInit(bubbles: Boolean? = false,
cancelable: Boolean? = false, composed: Boolean? = false): EventInit {\r\n val o = js("{}")\r\n\r\n o["bubbles"] = bubbles\r\n o["cancelable"] = cancelable\r\n o["composed"] = composed\r\n\r\n return

```

`o` Exposes the JavaScript `[CustomEvent]`(<https://developer.mozilla.org/en/docs/Web/API/CustomEvent>) to Kotlin `*^` public external open class `CustomEvent`(type: String, eventInitDict: CustomEventInit = definedExternally) : Event {  
 open val detail: Any?  
 fun initCustomEvent(type: String, bubbles: Boolean, cancelable: Boolean, detail: Any?): Unit  
} Exposes the JavaScript `[CustomEventInit]` to Kotlin `*^` public external interface `CustomEventInit` : `EventInit` {  
 var detail: Any? /\* = null \*/  
 get() = definedExternally  
 set(value) = definedExternally  
} Exposes the JavaScript `[EventListenerOptions]` to Kotlin `*^` public external interface `EventListenerOptions` {  
 var capture: Boolean? /\* = false \*/  
 get() = definedExternally  
 set(value) = definedExternally  
} Exposes the JavaScript `[AddEventListenerOptions]` to Kotlin `*^` public external interface `AddEventListenerOptions` : `EventListenerOptions` {  
 var passive: Boolean? /\* = false \*/  
 get() = definedExternally  
 set(value) = definedExternally  
 var once: Boolean? /\* = false \*/  
 get() = definedExternally  
 set(value) = definedExternally  
} Exposes the JavaScript `[ParentNode]`(<https://developer.mozilla.org/en/docs/Web/API/ParentNode>) to Kotlin `*^` public external interface `ParentNode` {  
 val children: HTMLCollection  
 val firstElementChild: Element?  
 val lastElementChild: Element?  
 val childElementCount: Int  
 fun prepend(vararg nodes: dynamic): Unit  
 fun append(vararg nodes: dynamic): Unit  
 fun querySelector(selectors: String): Element?  
 fun querySelectorAll(selectors: String): NodeList  
} Exposes the JavaScript `[NonDocumentTypeChildNode]`(<https://developer.mozilla.org/en/docs/Web/API/NonDocumentTypeChildNode>) to Kotlin `*^` public external interface `NonDocumentTypeChildNode` {  
 val previousElementSibling: Element?  
 val nextElementSibling: Element?  
} Exposes the JavaScript `[ChildNode]`(<https://developer.mozilla.org/en/docs/Web/API/ChildNode>) to Kotlin `*^` public external interface `ChildNode` {  
 fun before(vararg nodes: dynamic): Unit  
 fun after(vararg nodes: dynamic): Unit  
 fun replaceWith(vararg nodes: dynamic): Unit  
 fun remove(): Unit  
} Exposes the JavaScript `[Slotable]`(<https://developer.mozilla.org/en/docs/Web/API/Slotable>) to Kotlin `*^` public external interface `Slotable` {  
 val assignedSlot: HTMLSlotElement?  
} Exposes the JavaScript `[NodeList]`(<https://developer.mozilla.org/en/docs/Web/API/NodeList>) to Kotlin `*^` public external abstract class `NodeList` : `ItemArrayLike`<Node> {  
 override val length: Int  
 override fun item(index: Int): Node?  
} Exposes the JavaScript `[HTMLCollection]`(<https://developer.mozilla.org/en/docs/Web/API/HTMLCollection>) to Kotlin `*^` public external abstract class `HTMLCollection` : `UnionElementOrHTMLCollection`, `ItemArrayLike`<Element> {  
 override val length: Int  
 override fun item(index: Int): Element?  
 fun namedItem(name: String): Element?  
} Exposes the JavaScript `[MutationObserver]`(<https://developer.mozilla.org/en/docs/Web/API/MutationObserver>) to Kotlin `*^` public external open class `MutationObserver`(callback: (Array<MutationRecord>, MutationObserver) -> Unit) {  
 fun observe(target: Node, options: MutationObserverInit = definedExternally): Unit  
 fun disconnect(): Unit  
}





```

external open class XMLDocument : Document {
 override fun getElementById(elementId: String): Element?
 override fun prepend(vararg nodes: dynamic): Unit
 override fun append(vararg nodes: dynamic): Unit
 override fun querySelector(selectors: String): Element?
 override fun querySelectorAll(selectors: String): NodeList
 override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
 override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
 override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
 override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint
}

public external interface ElementCreationOptions {
 @JsName("is") var is_: String?
 get() = definedExternally
 set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ElementCreationOptions(is_: String? = undefined): ElementCreationOptions {
 val o = js("{}")
 o["is"] = is_
 return o
}

/** Exposes the JavaScript [DOMImplementation](https://developer.mozilla.org/en/docs/Web/API/DOMImplementation) to Kotlin */
public external abstract class DOMImplementation {
 fun createDocumentType(qualifiedName: String, publicId: String, systemId: String): DocumentType
 fun createDocument(namespace: String?, qualifiedName: String, doctype: DocumentType? = definedExternally): XMLDocument
 fun createHTMLDocument(title: String = definedExternally): Document
 fun hasFeature(): Boolean
}

/** Exposes the JavaScript [DocumentType](https://developer.mozilla.org/en/docs/Web/API/DocumentType) to Kotlin */
public external abstract class DocumentType : Node, ChildNode {
 open val name: String
 open val publicId: String
 open val systemId: String
}

/** Exposes the JavaScript [DocumentFragment](https://developer.mozilla.org/en/docs/Web/API/DocumentFragment) to Kotlin */
public external open class DocumentFragment : Node, NonElementParentNode, ParentNode {
 override val children: HTMLCollection
 override val firstElementChild: Element?
 override val lastElementChild: Element?
 override val childElementCount: Int
 override fun getElementById(elementId: String): Element?
 override fun prepend(vararg nodes: dynamic): Unit
 override fun append(vararg nodes: dynamic): Unit
 override fun querySelector(selectors: String): Element?
 override fun querySelectorAll(selectors: String): NodeList
}

/** Exposes the JavaScript [ShadowRoot](https://developer.mozilla.org/en/docs/Web/API/ShadowRoot) to Kotlin */
public external open class ShadowRoot : DocumentFragment, DocumentOrShadowRoot {
 open val mode: ShadowRootMode
 open val host: Element
 override val fullscreenElement: Element?
 override fun getElementById(elementId: String): Element?
 override fun prepend(vararg nodes: dynamic): Unit
 override fun append(vararg nodes: dynamic): Unit
 override fun querySelector(selectors: String): Element?
 override fun querySelectorAll(selectors: String): NodeList
}

/** Exposes the JavaScript [Element](https://developer.mozilla.org/en/docs/Web/API/Element) to Kotlin */
public external abstract class Element : Node, ParentNode, NonDocumentTypeChildNode, ChildNode, Slotable, GeometryUtils, UnionElementOrProcessingInstruction, UnionElementOrHTMLCollection, UnionElementOrRadioNodeList, UnionElementOrMouseEvent {
 open var innerHTML: String
 open var outerHTML: String
 open val namespaceURI: String?
 open val prefix: String?
 open val localName: String
 open val tagName: String
 open var id: String
 open var className: String
 open val classList: DOMTokenList
 open var slot: String
 open val attributes: NamedNodeMap
 open val shadowRoot: ShadowRoot?
 open var scrollTop: Double
 open var scrollLeft: Double
 open val scrollWidth: Int
 open val scrollHeight: Int
 open val clientTop: Int
 open val clientLeft: Int
 open val clientWidth: Int
 open val clientHeight: Int
 fun requestFullscreen(): Promise<Unit>
 fun setPointerCapture(pointerId: Int): Unit
 fun releasePointerCapture(pointerId: Int): Unit
 fun hasPointerCapture(pointerId: Int): Boolean
 fun insertAdjacentHTML(position: String, text: String): Unit
 fun hasAttributes(): Boolean
 fun getAttributeNames(): Array<String>
 fun getAttribute(qualifiedName: String): String?
 fun getAttributeNS(namespace: String?, localName: String): String?
 fun setAttribute(qualifiedName: String, value:

```

```

String): Unit\r\n fun setAttributeNS(namespace: String?, qualifiedName: String, value: String): Unit\r\n fun
removeAttribute(qualifiedName: String): Unit\r\n fun removeAttributeNS(namespace: String?, localName:
String): Unit\r\n fun hasAttribute(qualifiedName: String): Boolean\r\n fun hasAttributeNS(namespace: String?,
localName: String): Boolean\r\n fun getAttributeNode(qualifiedName: String): Attr?\r\n fun
getAttributeNodeNS(namespace: String?, localName: String): Attr?\r\n fun setAttributeNode(attr: Attr): Attr?\r\n
fun setAttributeNodeNS(attr: Attr): Attr?\r\n fun removeAttributeNode(attr: Attr): Attr\r\n fun
attachShadow(init: ShadowRootInit): ShadowRoot\r\n fun closest(selectors: String): Element?\r\n fun
matches(selectors: String): Boolean\r\n fun webkitMatchesSelector(selectors: String): Boolean\r\n fun
getElementsByTagName(qualifiedName: String): HTMLCollection\r\n fun
getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection\r\n fun
getElementsByClassName(classNames: String): HTMLCollection\r\n fun insertAdjacentElement(where: String,
element: Element): Element?\r\n fun insertAdjacentText(where: String, data: String): Unit\r\n fun
getClientRects(): Array<DOMRect>\r\n fun getBoundingClientRect(): DOMRect\r\n fun scrollIntoView():
Unit\r\n fun scrollIntoView(arg: dynamic): Unit\r\n fun scroll(options: ScrollToOptions = definedExternally):
Unit\r\n fun scroll(x: Double, y: Double): Unit\r\n fun scrollTo(options: ScrollToOptions = definedExternally):
Unit\r\n fun scrollTo(x: Double, y: Double): Unit\r\n fun scrollBy(options: ScrollToOptions =
definedExternally): Unit\r\n fun scrollBy(x: Double, y: Double): Unit\r\n\r\npublic external interface
ShadowRootInit {\r\n var mode: ShadowRootMode?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ShadowRootInit(mode:
ShadowRootMode?): ShadowRootInit {\r\n val o = js(\"({})\")\r\n\r\n o[\"mode\"] = mode\r\n\r\n return
o\r\n}\r\n\r\n\r\n**\r\n * Exposes the JavaScript
[NamedNodeMap](https://developer.mozilla.org/en/docs/Web/API/NamedNodeMap) to Kotlin\r\n *^\r\n\r\npublic
external abstract class NamedNodeMap : ItemArrayLike<Attr> {\r\n override val length: Int\r\n override fun
item(index: Int): Attr?\r\n fun getNamedItem(qualifiedName: String): Attr?\r\n fun
getNamedItemNS(namespace: String?, localName: String): Attr?\r\n fun setNamedItem(attr: Attr): Attr?\r\n fun
setNamedItemNS(attr: Attr): Attr?\r\n fun removeNamedItem(qualifiedName: String): Attr\r\n fun
removeNamedItemNS(namespace: String?, localName: String): Attr\r\n}\r\n\r\n@kotlin.internal.InlineOnly inline
operator fun NamedNodeMap.get(index: Int): Attr? = asDynamic()[index]\r\n\r\n@kotlin.internal.InlineOnly inline
operator fun NamedNodeMap.get(qualifiedName: String): Attr? = asDynamic()[qualifiedName]\r\n\r\n\r\n**\r\n *
Exposes the JavaScript [Attr](https://developer.mozilla.org/en/docs/Web/API/Attr) to Kotlin\r\n *^\r\n\r\npublic
external abstract class Attr : Node {\r\n open val namespaceURI: String?\r\n open val prefix: String?\r\n open
val localName: String\r\n open val name: String\r\n open var value: String\r\n open val ownerElement:
Element?\r\n open val specified: Boolean\r\n}\r\n\r\n\r\n**\r\n * Exposes the JavaScript
[CharacterData](https://developer.mozilla.org/en/docs/Web/API/CharacterData) to Kotlin\r\n *^\r\n\r\npublic external
abstract class CharacterData : Node, NonDocumentTypeChildNode, ChildNode {\r\n open var data: String\r\n
open val length: Int\r\n fun substringData(offset: Int, count: Int): String\r\n fun appendData(data: String):
Unit\r\n fun insertData(offset: Int, data: String): Unit\r\n fun deleteData(offset: Int, count: Int): Unit\r\n fun
replaceData(offset: Int, count: Int, data: String): Unit\r\n}\r\n\r\n\r\n**\r\n * Exposes the JavaScript
[Text](https://developer.mozilla.org/en/docs/Web/API/Text) to Kotlin\r\n *^\r\n\r\npublic external open class Text(data:
String = definedExternally) : CharacterData, Slotable, GeometryUtils {\r\n open val wholeText: String\r\n
override val previousElementSibling: Element?\r\n override val nextElementSibling: Element?\r\n override val
assignedSlot: HTMLSlotElement?\r\n fun splitText(offset: Int): Text\r\n override fun before(vararg nodes:
dynamic): Unit\r\n override fun after(vararg nodes: dynamic): Unit\r\n override fun replaceWith(vararg nodes:
dynamic): Unit\r\n override fun remove(): Unit\r\n override fun getBoxQuads(options: BoxQuadOptions /* =
definedExternally */): Array<DOMQuad>\r\n override fun convertQuadFromNode(quad: dynamic, from:
dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\r\n override fun
convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* =
definedExternally */): DOMQuad\r\n override fun convertPointFromNode(point: DOMPointInit, from: dynamic,

```

options: ConvertCoordinateOptions /\* = definedExternally \*/): DOMPoint\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript [CDATASection](https://developer.mozilla.org/en/docs/Web/API/CDATASection) to Kotlin\r\n\*\r\n\r\npublic external open class CDATASection : Text {\r\n override fun before(vararg nodes: dynamic): Unit\r\n override fun after(vararg nodes: dynamic): Unit\r\n override fun replaceWith(vararg nodes: dynamic): Unit\r\n override fun remove(): Unit\r\n override fun getBoxQuads(options: BoxQuadOptions /\* = definedExternally \*/): Array<DOMQuad>\r\n override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /\* = definedExternally \*/): DOMQuad\r\n override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /\* = definedExternally \*/): DOMQuad\r\n override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /\* = definedExternally \*/): DOMPoint\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript [ProcessingInstruction](https://developer.mozilla.org/en/docs/Web/API/ProcessingInstruction) to Kotlin\r\n\*\r\n\r\npublic external abstract class ProcessingInstruction : CharacterData, LinkStyle, UnionElementOrProcessingInstruction {\r\n open val target: String\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript [Comment](https://developer.mozilla.org/en/docs/Web/API/Comment) to Kotlin\r\n\*\r\n\r\npublic external open class Comment(data: String = definedExternally) : CharacterData {\r\n override val previousElementSibling: Element?\r\n override val nextElementSibling: Element?\r\n override fun before(vararg nodes: dynamic): Unit\r\n override fun after(vararg nodes: dynamic): Unit\r\n override fun replaceWith(vararg nodes: dynamic): Unit\r\n override fun remove(): Unit\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript [Range](https://developer.mozilla.org/en/docs/Web/API/Range) to Kotlin\r\n\*\r\n\r\npublic external open class Range {\r\n open val startContainer: Node\r\n open val startOffset: Int\r\n open val endContainer: Node\r\n open val endOffset: Int\r\n open val collapsed: Boolean\r\n open val commonAncestorContainer: Node\r\n fun createContextualFragment(fragment: String): DocumentFragment\r\n fun setStart(node: Node, offset: Int): Unit\r\n fun setEnd(node: Node, offset: Int): Unit\r\n fun setStartBefore(node: Node): Unit\r\n fun setStartAfter(node: Node): Unit\r\n fun setEndBefore(node: Node): Unit\r\n fun setEndAfter(node: Node): Unit\r\n fun collapse(toStart: Boolean = definedExternally): Unit\r\n fun selectNode(node: Node): Unit\r\n fun selectNodeContents(node: Node): Unit\r\n fun compareBoundaryPoints(how: Short, sourceRange: Range): Short\r\n fun deleteContents(): Unit\r\n fun extractContents(): DocumentFragment\r\n fun cloneContents(): DocumentFragment\r\n fun insertNode(node: Node): Unit\r\n fun surroundContents(newParent: Node): Unit\r\n fun cloneRange(): Range\r\n fun detach(): Unit\r\n fun isPointInRange(node: Node, offset: Int): Boolean\r\n fun comparePoint(node: Node, offset: Int): Short\r\n fun intersectsNode(node: Node): Boolean\r\n fun getClientRects(): Array<DOMRect>\r\n fun getBoundingClientRect(): DOMRect\r\n\r\n companion object {\r\n val START\_TO\_START: Short\r\n val START\_TO\_END: Short\r\n val END\_TO\_END: Short\r\n val END\_TO\_START: Short\r\n }\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript [NodeIterator](https://developer.mozilla.org/en/docs/Web/API/NodeIterator) to Kotlin\r\n\*\r\n\r\npublic external abstract class NodeIterator {\r\n open val root: Node\r\n open val referenceNode: Node\r\n open val pointerBeforeReferenceNode: Boolean\r\n open val whatToShow: Int\r\n open val filter: NodeFilter?\r\n fun nextNode(): Node?\r\n fun previousNode(): Node?\r\n fun detach(): Unit\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript [TreeWalker](https://developer.mozilla.org/en/docs/Web/API/TreeWalker) to Kotlin\r\n\*\r\n\r\npublic external abstract class TreeWalker {\r\n open val root: Node\r\n open val whatToShow: Int\r\n open val filter: NodeFilter?\r\n open var currentNode: Node\r\n fun parentNode(): Node?\r\n fun firstChild(): Node?\r\n fun lastChild(): Node?\r\n fun previousSibling(): Node?\r\n fun nextSibling(): Node?\r\n fun previousNode(): Node?\r\n fun nextNode(): Node?\r\n}\r\n\r\n\*\*\r\n \* Exposes the JavaScript [NodeFilter](https://developer.mozilla.org/en/docs/Web/API/NodeFilter) to Kotlin\r\n\*\r\n\r\npublic external interface NodeFilter {\r\n fun acceptNode(node: Node): Short\r\n\r\n companion object {\r\n val FILTER\_ACCEPT: Short\r\n val FILTER\_REJECT: Short\r\n val FILTER\_SKIP: Short\r\n val SHOW\_ALL: Int\r\n val SHOW\_ELEMENT: Int\r\n val SHOW\_ATTRIBUTE: Int\r\n val SHOW\_TEXT: Int\r\n val SHOW\_CDATA\_SECTION: Int\r\n val SHOW\_ENTITY\_REFERENCE: Int\r\n val SHOW\_ENTITY: Int\r\n val SHOW\_PROCESSING\_INSTRUCTION: Int\r\n val SHOW\_COMMENT: Int\r\n val





```

set(value) = definedExternally\r\n var top: Double?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ScrollToOptions(left: Double? =
undefined, top: Double? = undefined, behavior: ScrollBehavior? = ScrollBehavior.AUTO): ScrollToOptions {\r\n
val o = js("{}")\r\n\r\n o["left"] = left\r\n o["top"] = top\r\n o["behavior"] = behavior\r\n\r\n return
o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[MediaQueryList](https://developer.mozilla.org/en/docs/Web/API/MediaQueryList) to Kotlin\r\n *^\r\npublic
external abstract class MediaQueryList : EventTarget {\r\n open val media: String\r\n open val matches:
Boolean\r\n open var onchange: ((Event) -> dynamic)?\r\n fun addListener(listener: EventListener?): Unit\r\n
fun addListener(listener: ((Event) -> Unit)?): Unit\r\n fun removeListener(listener: EventListener?): Unit\r\n
fun removeListener(listener: ((Event) -> Unit)?): Unit\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[MediaQueryListEvent](https://developer.mozilla.org/en/docs/Web/API/MediaQueryListEvent) to Kotlin\r\n
*^\r\npublic external open class MediaQueryListEvent(type: String, eventInitDict: MediaQueryListEventInit =
definedExternally) : Event {\r\n open val media: String\r\n open val matches: Boolean\r\n}\r\n\r\npublic external
interface MediaQueryListEventInit : EventInit {\r\n var media: String? /* = "" */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var matches: Boolean? /* = false */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
MediaQueryListEventInit(media: String? = "", matches: Boolean? = false, bubbles: Boolean? = false, cancelable:
Boolean? = false, composed: Boolean? = false): MediaQueryListEventInit {\r\n val o = js("{}")\r\n\r\n
o["media"] = media\r\n o["matches"] = matches\r\n o["bubbles"] = bubbles\r\n o["cancelable"] =
cancelable\r\n o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[Screen](https://developer.mozilla.org/en/docs/Web/API/Screen) to Kotlin\r\n *^\r\npublic external abstract class
Screen {\r\n open val availWidth: Int\r\n open val availHeight: Int\r\n open val width: Int\r\n open val height:
Int\r\n open val colorDepth: Int\r\n open val pixelDepth: Int\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[CaretPosition](https://developer.mozilla.org/en/docs/Web/API/CaretPosition) to Kotlin\r\n *^\r\npublic external
abstract class CaretPosition {\r\n open val offsetNode: Node\r\n open val offset: Int\r\n fun getClientRect():
DOMRect?\r\n}\r\n\r\npublic external interface ScrollIntoViewOptions : ScrollOptions {\r\n var block:
ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */\r\n get() = definedExternally\r\n set(value)
= definedExternally\r\n var inline: ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
ScrollIntoViewOptions(block: ScrollLogicalPosition? = ScrollLogicalPosition.CENTER, inline:
ScrollLogicalPosition? = ScrollLogicalPosition.CENTER, behavior: ScrollBehavior? = ScrollBehavior.AUTO):
ScrollIntoViewOptions {\r\n val o = js("{}")\r\n\r\n o["block"] = block\r\n o["inline"] = inline\r\n
o["behavior"] = behavior\r\n\r\n return o\r\n}\r\n\r\npublic external interface BoxQuadOptions {\r\n var box:
CSSBoxType? /* = CSSBoxType.BORDER */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var relativeTo: dynamic\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun BoxQuadOptions(box: CSSBoxType?
= CSSBoxType.BORDER, relativeTo: dynamic = undefined): BoxQuadOptions {\r\n val o = js("{}")\r\n\r\n
o["box"] = box\r\n o["relativeTo"] = relativeTo\r\n\r\n return o\r\n}\r\n\r\npublic external interface
ConvertCoordinateOptions {\r\n var fromBox: CSSBoxType? /* = CSSBoxType.BORDER */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var toBox: CSSBoxType? /* =
CSSBoxType.BORDER */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ConvertCoordinateOptions(fromBox:
CSSBoxType? = CSSBoxType.BORDER, toBox: CSSBoxType? = CSSBoxType.BORDER):
ConvertCoordinateOptions {\r\n val o = js("{}")\r\n\r\n o["fromBox"] = fromBox\r\n o["toBox"] =
toBox\r\n\r\n return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[GeometryUtils](https://developer.mozilla.org/en/docs/Web/API/GeometryUtils) to Kotlin\r\n *^\r\npublic external
interface GeometryUtils {\r\n fun getBoxQuads(options: BoxQuadOptions = definedExternally):
Array<DOMQuad>\r\n fun convertQuadFromNode(quad: dynamic, from: dynamic, options:

```

```

ConvertCoordinateOptions = definedExternally): DOMQuad\r\n fun convertRectFromNode(rect:
DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMQuad\r\n fun
convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions =
definedExternally): DOMPoint\r\n}\r\n\r\npublic external @marker interface UnionElementOrProcessingInstruction
{\r\n}\r\n\r\npublic external @marker interface UnionElementOrHTMLCollection {\r\n}\r\n\r\npublic external
@marker interface UnionElementOrRadioNodeList {\r\n}\r\n\r\npublic external @marker interface
UnionHTMLOptGroupElementOrHTMLOptionElement {\r\n}\r\n\r\npublic external @marker interface
UnionAudioTrackOrTextTrackOrVideoTrack {\r\n}\r\n\r\npublic external @marker interface
UnionElementOrMouseEvent {\r\n}\r\n\r\npublic external @marker interface UnionMessagePortOrWindow
{\r\n}\r\n\r\npublic external @marker interface UnionMessagePortOrServiceWorker {\r\n}\r\n\r\npublic external
@marker interface HTMLOrSVGScriptElement {\r\n}\r\n\r\npublic external @marker interface RenderingContext
{\r\n}\r\n\r\npublic external @marker interface HTMLOrSVGImageElement {\r\n}\r\n\r\n/* please, don't implement
this interface! *\r\n\r\npublic external interface DocumentReadyState {\r\n companion object\r\n}\r\n\r\npublic inline val
DocumentReadyState.Companion.LOADING: DocumentReadyState get() =
"loading".asDynamic().unsafeCast<DocumentReadyState>()\r\n\r\npublic inline val
DocumentReadyState.Companion.INTERACTIVE: DocumentReadyState get() =
"interactive".asDynamic().unsafeCast<DocumentReadyState>()\r\n\r\npublic inline val
DocumentReadyState.Companion.COMPLETE: DocumentReadyState get() =
"complete".asDynamic().unsafeCast<DocumentReadyState>()\r\n\r\n\r\n/* please, don't implement this interface!
*\r\n\r\npublic external interface CanPlayTypeResult {\r\n companion object\r\n}\r\n\r\npublic inline val
CanPlayTypeResult.Companion.EMPTY: CanPlayTypeResult get() =
"".asDynamic().unsafeCast<CanPlayTypeResult>()\r\n\r\npublic inline val CanPlayTypeResult.Companion.MAYBE:
CanPlayTypeResult get() = "maybe".asDynamic().unsafeCast<CanPlayTypeResult>()\r\n\r\npublic inline val
CanPlayTypeResult.Companion.PROBABLY: CanPlayTypeResult get() =
"probably".asDynamic().unsafeCast<CanPlayTypeResult>()\r\n\r\n\r\n/* please, don't implement this interface!
*\r\n\r\npublic external interface TextTrackMode {\r\n companion object\r\n}\r\n\r\npublic inline val
TextTrackMode.Companion.DISABLED: TextTrackMode get() =
"disabled".asDynamic().unsafeCast<TextTrackMode>()\r\n\r\npublic inline val
TextTrackMode.Companion.HIDDEN: TextTrackMode get() =
"hidden".asDynamic().unsafeCast<TextTrackMode>()\r\n\r\npublic inline val
TextTrackMode.Companion.SHOWING: TextTrackMode get() =
"showing".asDynamic().unsafeCast<TextTrackMode>()\r\n\r\n\r\n/* please, don't implement this interface!
*\r\n\r\npublic external interface TextTrackKind {\r\n companion object\r\n}\r\n\r\npublic inline val
TextTrackKind.Companion.SUBTITLES: TextTrackKind get() =
"subtitles".asDynamic().unsafeCast<TextTrackKind>()\r\n\r\npublic inline val
TextTrackKind.Companion.CAPTIONS: TextTrackKind get() =
"captions".asDynamic().unsafeCast<TextTrackKind>()\r\n\r\npublic inline val
TextTrackKind.Companion.DESCRPTIONS: TextTrackKind get() =
"descriptions".asDynamic().unsafeCast<TextTrackKind>()\r\n\r\npublic inline val
TextTrackKind.Companion.CHAPTERS: TextTrackKind get() =
"chapters".asDynamic().unsafeCast<TextTrackKind>()\r\n\r\npublic inline val
TextTrackKind.Companion.METADATA: TextTrackKind get() =
"metadata".asDynamic().unsafeCast<TextTrackKind>()\r\n\r\n\r\n/* please, don't implement this interface!
*\r\n\r\npublic external interface SelectionMode {\r\n companion object\r\n}\r\n\r\npublic inline val
SelectionMode.Companion.SELECT: SelectionMode get() =
"select".asDynamic().unsafeCast<SelectionMode>()\r\n\r\npublic inline val SelectionMode.Companion.START:
SelectionMode get() = "start".asDynamic().unsafeCast<SelectionMode>()\r\n\r\npublic inline val
SelectionMode.Companion.END: SelectionMode get() =

```

```

"end".asDynamic().unsafeCast<SelectionMode>()\r\npublic inline val SelectionMode.Companion.PRESERVE:
SelectionMode get() = "preserve".asDynamic().unsafeCast<SelectionMode>()\r\n\r\n/* please, don't implement
this interface! */\r\npublic external interface CanvasFillRule {\r\n companion object\r\n}\r\npublic inline val
CanvasFillRule.Companion.NONZERO: CanvasFillRule get() =
"nonzero".asDynamic().unsafeCast<CanvasFillRule>()\r\npublic inline val
CanvasFillRule.Companion.EVENODD: CanvasFillRule get() =
"evenodd".asDynamic().unsafeCast<CanvasFillRule>()\r\n\r\n/* please, don't implement this interface!
*/\r\npublic external interface ImageSmoothingQuality {\r\n companion object\r\n}\r\npublic inline val
ImageSmoothingQuality.Companion.LOW: ImageSmoothingQuality get() =
"low".asDynamic().unsafeCast<ImageSmoothingQuality>()\r\npublic inline val
ImageSmoothingQuality.Companion.MEDIUM: ImageSmoothingQuality get() =
"medium".asDynamic().unsafeCast<ImageSmoothingQuality>()\r\npublic inline val
ImageSmoothingQuality.Companion.HIGH: ImageSmoothingQuality get() =
"high".asDynamic().unsafeCast<ImageSmoothingQuality>()\r\n\r\n/* please, don't implement this interface!
*/\r\npublic external interface CanvasLineCap {\r\n companion object\r\n}\r\npublic inline val
CanvasLineCap.Companion.BUTT: CanvasLineCap get() =
"butt".asDynamic().unsafeCast<CanvasLineCap>()\r\npublic inline val CanvasLineCap.Companion.ROUND:
CanvasLineCap get() = "round".asDynamic().unsafeCast<CanvasLineCap>()\r\npublic inline val
CanvasLineCap.Companion.SQUARE: CanvasLineCap get() =
"square".asDynamic().unsafeCast<CanvasLineCap>()\r\n\r\n/* please, don't implement this interface! */\r\npublic
external interface CanvasLineJoin {\r\n companion object\r\n}\r\npublic inline val
CanvasLineJoin.Companion.ROUND: CanvasLineJoin get() =
"round".asDynamic().unsafeCast<CanvasLineJoin>()\r\npublic inline val CanvasLineJoin.Companion.BEVEL:
CanvasLineJoin get() = "bevel".asDynamic().unsafeCast<CanvasLineJoin>()\r\npublic inline val
CanvasLineJoin.Companion.MITER: CanvasLineJoin get() =
"miter".asDynamic().unsafeCast<CanvasLineJoin>()\r\n\r\n/* please, don't implement this interface! */\r\npublic
external interface CanvasTextAlign {\r\n companion object\r\n}\r\npublic inline val
CanvasTextAlign.Companion.START: CanvasTextAlign get() =
"start".asDynamic().unsafeCast<CanvasTextAlign>()\r\npublic inline val CanvasTextAlign.Companion.END:
CanvasTextAlign get() = "end".asDynamic().unsafeCast<CanvasTextAlign>()\r\npublic inline val
CanvasTextAlign.Companion.LEFT: CanvasTextAlign get() =
"left".asDynamic().unsafeCast<CanvasTextAlign>()\r\npublic inline val CanvasTextAlign.Companion.RIGHT:
CanvasTextAlign get() = "right".asDynamic().unsafeCast<CanvasTextAlign>()\r\npublic inline val
CanvasTextAlign.Companion.CENTER: CanvasTextAlign get() =
"center".asDynamic().unsafeCast<CanvasTextAlign>()\r\n\r\n/* please, don't implement this interface! */\r\npublic
external interface CanvasTextBaseline {\r\n companion object\r\n}\r\npublic inline val
CanvasTextBaseline.Companion.TOP: CanvasTextBaseline get() =
"top".asDynamic().unsafeCast<CanvasTextBaseline>()\r\npublic inline val
CanvasTextBaseline.Companion.HANGING: CanvasTextBaseline get() =
"hanging".asDynamic().unsafeCast<CanvasTextBaseline>()\r\npublic inline val
CanvasTextBaseline.Companion.MIDDLE: CanvasTextBaseline get() =
"middle".asDynamic().unsafeCast<CanvasTextBaseline>()\r\npublic inline val
CanvasTextBaseline.Companion.ALPHABETIC: CanvasTextBaseline get() =
"alphabetic".asDynamic().unsafeCast<CanvasTextBaseline>()\r\npublic inline val
CanvasTextBaseline.Companion.IDEOGRAPHIC: CanvasTextBaseline get() =
"ideographic".asDynamic().unsafeCast<CanvasTextBaseline>()\r\npublic inline val
CanvasTextBaseline.Companion.BOTTOM: CanvasTextBaseline get() =
"bottom".asDynamic().unsafeCast<CanvasTextBaseline>()\r\n\r\n/* please, don't implement this interface!

```



```

*\/r\npublic external interface CanvasDirection {\/r\n companion object\/r\n}\/r\npublic inline val
CanvasDirection.Companion.LTR: CanvasDirection get() =
"ltr".asDynamic().unsafeCast<CanvasDirection>()\/r\npublic inline val CanvasDirection.Companion.RTL:
CanvasDirection get() = "rtl".asDynamic().unsafeCast<CanvasDirection>()\/r\npublic inline val
CanvasDirection.Companion.INHERIT: CanvasDirection get() =
"inherit".asDynamic().unsafeCast<CanvasDirection>()\/r\n\/r\n/* please, don't implement this interface! */\/r\npublic
external interface ScrollRestoration {\/r\n companion object\/r\n}\/r\npublic inline val
ScrollRestoration.Companion.AUTO: ScrollRestoration get() =
"auto".asDynamic().unsafeCast<ScrollRestoration>()\/r\npublic inline val
ScrollRestoration.Companion.MANUAL: ScrollRestoration get() =
"manual".asDynamic().unsafeCast<ScrollRestoration>()\/r\n\/r\n/* please, don't implement this interface!
*/\/r\npublic external interface ImageOrientation {\/r\n companion object\/r\n}\/r\npublic inline val
ImageOrientation.Companion.NONE: ImageOrientation get() =
"none".asDynamic().unsafeCast<ImageOrientation>()\/r\npublic inline val ImageOrientation.Companion.FLIPY:
ImageOrientation get() = "flipY".asDynamic().unsafeCast<ImageOrientation>()\/r\n\/r\n/* please, don't implement
this interface! */\/r\npublic external interface PremultiplyAlpha {\/r\n companion object\/r\n}\/r\npublic inline val
PremultiplyAlpha.Companion.NONE: PremultiplyAlpha get() =
"none".asDynamic().unsafeCast<PremultiplyAlpha>()\/r\npublic inline val
PremultiplyAlpha.Companion.PREMULTIPLY: PremultiplyAlpha get() =
"premultiply".asDynamic().unsafeCast<PremultiplyAlpha>()\/r\npublic inline val
PremultiplyAlpha.Companion.DEFAULT: PremultiplyAlpha get() =
"default".asDynamic().unsafeCast<PremultiplyAlpha>()\/r\n\/r\n/* please, don't implement this interface!
*/\/r\npublic external interface ColorSpaceConversion {\/r\n companion object\/r\n}\/r\npublic inline val
ColorSpaceConversion.Companion.NONE: ColorSpaceConversion get() =
"none".asDynamic().unsafeCast<ColorSpaceConversion>()\/r\npublic inline val
ColorSpaceConversion.Companion.DEFAULT: ColorSpaceConversion get() =
"default".asDynamic().unsafeCast<ColorSpaceConversion>()\/r\n\/r\n/* please, don't implement this interface!
*/\/r\npublic external interface ResizeQuality {\/r\n companion object\/r\n}\/r\npublic inline val
ResizeQuality.Companion.PIXELATED: ResizeQuality get() =
"pixelated".asDynamic().unsafeCast<ResizeQuality>()\/r\npublic inline val ResizeQuality.Companion.LOW:
ResizeQuality get() = "low".asDynamic().unsafeCast<ResizeQuality>()\/r\npublic inline val
ResizeQuality.Companion.MEDIUM: ResizeQuality get() =
"medium".asDynamic().unsafeCast<ResizeQuality>()\/r\npublic inline val ResizeQuality.Companion.HIGH:
ResizeQuality get() = "high".asDynamic().unsafeCast<ResizeQuality>()\/r\n\/r\n/* please, don't implement this
interface! */\/r\npublic external interface BinaryType {\/r\n companion object\/r\n}\/r\npublic inline val
BinaryType.Companion.BLOB: BinaryType get() = "blob".asDynamic().unsafeCast<BinaryType>()\/r\npublic
inline val BinaryType.Companion.ARRAYBUFFER: BinaryType get() =
"arraybuffer".asDynamic().unsafeCast<BinaryType>()\/r\n\/r\n/* please, don't implement this interface! */\/r\npublic
external interface WorkerType {\/r\n companion object\/r\n}\/r\npublic inline val
WorkerType.Companion.CLASSIC: WorkerType get() =
"classic".asDynamic().unsafeCast<WorkerType>()\/r\npublic inline val WorkerType.Companion.MODULE:
WorkerType get() = "module".asDynamic().unsafeCast<WorkerType>()\/r\n\/r\n/* please, don't implement this
interface! */\/r\npublic external interface ShadowRootMode {\/r\n companion object\/r\n}\/r\npublic inline val
ShadowRootMode.Companion.OPEN: ShadowRootMode get() =
"open".asDynamic().unsafeCast<ShadowRootMode>()\/r\npublic inline val
ShadowRootMode.Companion.CLOSED: ShadowRootMode get() =
"closed".asDynamic().unsafeCast<ShadowRootMode>()\/r\n\/r\n/* please, don't implement this interface!
*/\/r\npublic external interface ScrollBehavior {\/r\n companion object\/r\n}\/r\npublic inline val

```

```

ScrollBehavior.Companion.AUTO: ScrollBehavior get() =
\"auto\".asDynamic().unsafeCast<ScrollBehavior>()\r\npublic inline val ScrollBehavior.Companion.INSTANT:
ScrollBehavior get() = \"instant\".asDynamic().unsafeCast<ScrollBehavior>()\r\npublic inline val
ScrollBehavior.Companion.SMOOTH: ScrollBehavior get() =
\"smooth\".asDynamic().unsafeCast<ScrollBehavior>()\r\n\r\n/* please, don't implement this interface! */\r\npublic
external interface ScrollLogicalPosition {\r\n companion object\r\n}\r\npublic inline val
ScrollLogicalPosition.Companion.START: ScrollLogicalPosition get() =
\"start\".asDynamic().unsafeCast<ScrollLogicalPosition>()\r\npublic inline val
ScrollLogicalPosition.Companion.CENTER: ScrollLogicalPosition get() =
\"center\".asDynamic().unsafeCast<ScrollLogicalPosition>()\r\npublic inline val
ScrollLogicalPosition.Companion.END: ScrollLogicalPosition get() =
\"end\".asDynamic().unsafeCast<ScrollLogicalPosition>()\r\npublic inline val
ScrollLogicalPosition.Companion.NEAREST: ScrollLogicalPosition get() =
\"nearest\".asDynamic().unsafeCast<ScrollLogicalPosition>()\r\n\r\n/* please, don't implement this interface!
*/\r\npublic external interface CSSBoxType {\r\n companion object\r\n}\r\npublic inline val
CSSBoxType.Companion.MARGIN: CSSBoxType get() =
\"margin\".asDynamic().unsafeCast<CSSBoxType>()\r\npublic inline val CSSBoxType.Companion.BORDER:
CSSBoxType get() = \"border\".asDynamic().unsafeCast<CSSBoxType>()\r\npublic inline val
CSSBoxType.Companion.PADDING: CSSBoxType get() =
\"padding\".asDynamic().unsafeCast<CSSBoxType>()\r\npublic inline val CSSBoxType.Companion.CONTENT:
CSSBoxType get() = \"content\".asDynamic().unsafeCast<CSSBoxType>()\r\n\r\n\", \"/*\r\n * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n * \r\n\r\n// NOTE: THIS FILE IS AUTO-
GENERATED, DO NOT EDIT!\r\n// See libraries/tools/idl2k for
details\r\n\r\n@file:Suppress(\"NESTED_CLASS_IN_EXTERNAL_INTERFACE\")\r\npackage
org.w3c.fetch\r\n\r\nimport kotlin.js.*\r\nimport org.khronos.webgl.*\r\nimport org.w3c.css.masking.*\r\nimport
org.w3c.dom.*\r\nimport org.w3c.dom.clipboard.*\r\nimport org.w3c.dom.css.*\r\nimport
org.w3c.dom.events.*\r\nimport org.w3c.dom.mediacapture.*\r\nimport org.w3c.dom.parsing.*\r\nimport
org.w3c.dom.pointerevents.*\r\nimport org.w3c.dom.svg.*\r\nimport org.w3c.dom.url.*\r\nimport
org.w3c.files.*\r\nimport org.w3c.notifications.*\r\nimport org.w3c.performance.*\r\nimport
org.w3c.workers.*\r\nimport org.w3c.xhr.*\r\n\r\n/* Exposes the JavaScript
[Headers](https://developer.mozilla.org/en/docs/Web/API/Headers) to Kotlin\r\n */\r\npublic external open class
Headers(init: dynamic = definedExternally) {\r\n fun append(name: String, value: String): Unit\r\n fun
delete(name: String): Unit\r\n fun get(name: String): String?\r\n fun has(name: String): Boolean\r\n fun
set(name: String, value: String): Unit\r\n}\r\n\r\n/* Exposes the JavaScript
[Body](https://developer.mozilla.org/en/docs/Web/API/Body) to Kotlin\r\n */\r\npublic external interface Body {\r\n
 val bodyUsed: Boolean\r\n fun arrayBuffer(): Promise<ArrayBuffer>\r\n fun blob(): Promise<Blob>\r\n fun
formData(): Promise<FormData>\r\n fun json(): Promise<Any?>\r\n fun text():
Promise<String>\r\n}\r\n\r\n/* Exposes the JavaScript
[Request](https://developer.mozilla.org/en/docs/Web/API/Request) to Kotlin\r\n */\r\npublic external open class
Request(input: dynamic, init: RequestInit = definedExternally) : Body {\r\n open val method: String\r\n open val
url: String\r\n open val headers: Headers\r\n open val type: RequestType\r\n open val destination:
RequestDestination\r\n open val referrer: String\r\n open val referrerPolicy: dynamic\r\n open val mode:
RequestMode\r\n open val credentials: RequestCredentials\r\n open val cache: RequestCache\r\n open val
redirect: RequestRedirect\r\n open val integrity: String\r\n open val keepalive: Boolean\r\n override val
bodyUsed: Boolean\r\n fun clone(): Request\r\n override fun arrayBuffer(): Promise<ArrayBuffer>\r\n
 override fun blob(): Promise<Blob>\r\n override fun formData(): Promise<FormData>\r\n override fun json():
Promise<Any?>\r\n override fun text(): Promise<String>\r\n}\r\n\r\npublic external interface RequestInit {\r\n

```

```

var method: String? \r\n get() = definedExternally \r\n set(value) = definedExternally \r\n var headers:
dynamic \r\n get() = definedExternally \r\n set(value) = definedExternally \r\n var body: dynamic \r\n
get() = definedExternally \r\n set(value) = definedExternally \r\n var referrer: String? \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n var referrerPolicy: dynamic \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n var mode: RequestMode? \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n var credentials: RequestCredentials? \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n var cache: RequestCache? \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n var redirect: RequestRedirect? \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n var integrity: String? \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n var keepalive: Boolean? \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n var window: Any? \r\n get() =
definedExternally \r\n set(value) = definedExternally \r\n} \r\n \r\n @kotlin.internal.InlineOnly \r\n public inline fun
RequestInit(method: String? = undefined, headers: dynamic = undefined, body: dynamic = undefined, referrer:
String? = undefined, referrerPolicy: dynamic = undefined, mode: RequestMode? = undefined, credentials:
RequestCredentials? = undefined, cache: RequestCache? = undefined, redirect: RequestRedirect? = undefined,
integrity: String? = undefined, keepalive: Boolean? = undefined, window: Any? = undefined): RequestInit { \r\n
val o = js("{}") \r\n \r\n o["method"] = method \r\n o["headers"] = headers \r\n o["body"] = body \r\n
o["referrer"] = referrer \r\n o["referrerPolicy"] = referrerPolicy \r\n o["mode"] = mode \r\n o["credentials"]
= credentials \r\n o["cache"] = cache \r\n o["redirect"] = redirect \r\n o["integrity"] = integrity \r\n
o["keepalive"] = keepalive \r\n o["window"] = window \r\n \r\n return o \r\n} \r\n \r\n /** Exposes the
JavaScript [Response](https://developer.mozilla.org/en/docs/Web/API/Response) to Kotlin \r\n * \r\n public external
open class Response(body: dynamic = definedExternally, init: ResponseInit = definedExternally) : Body { \r\n
open val type: ResponseType \r\n open val url: String \r\n open val redirected: Boolean \r\n open val status:
Short \r\n open val ok: Boolean \r\n open val statusText: String \r\n open val headers: Headers \r\n open val
body: dynamic \r\n open val trailer: Promise<Headers> \r\n override val bodyUsed: Boolean \r\n fun clone():
Response \r\n override fun arrayBuffer(): Promise<ArrayBuffer> \r\n override fun blob(): Promise<Blob> \r\n
override fun formData(): Promise<FormData> \r\n override fun json(): Promise<Any?> \r\n override fun text():
Promise<String> \r\n \r\n companion object { \r\n fun error(): Response \r\n fun redirect(url: String, status:
Short = definedExternally): Response \r\n } \r\n} \r\n \r\n public external interface ResponseInit { \r\n var status:
Short? /* = 200 */ \r\n get() = definedExternally \r\n set(value) = definedExternally \r\n var statusText:
String? /* = "OK" */ \r\n get() = definedExternally \r\n set(value) = definedExternally \r\n var headers:
dynamic \r\n get() = definedExternally \r\n set(value) =
definedExternally \r\n} \r\n \r\n @kotlin.internal.InlineOnly \r\n public inline fun ResponseInit(status: Short? = 200,
statusText: String? = "OK", headers: dynamic = undefined): ResponseInit { \r\n val o = js("{}") \r\n \r\n
o["status"] = status \r\n o["statusText"] = statusText \r\n o["headers"] = headers \r\n \r\n return
o \r\n} \r\n \r\n /** please, don't implement this interface! \r\n public external interface RequestType { \r\n companion
object \r\n} \r\n \r\n public inline val RequestType.Companion.EMPTY: RequestType get() =
"" \r\n \r\n public inline val RequestType.Companion.AUDIO: RequestType
get() = "audio" \r\n \r\n public inline val RequestType.Companion.FONT:
RequestType get() = "font" \r\n \r\n public inline val
RequestType.Companion.IMAGE: RequestType get() =
"image" \r\n \r\n public inline val RequestType.Companion.SCRIPT:
RequestType get() = "script" \r\n \r\n public inline val
RequestType.Companion.STYLE: RequestType get() =
"style" \r\n \r\n public inline val RequestType.Companion.TRACK:
RequestType get() = "track" \r\n \r\n public inline val
RequestType.Companion.VIDEO: RequestType get() =
"video" \r\n \r\n /** please, don't implement this interface! \r\n public

```

```

external interface RequestDestination {
 companion object
}
public inline val
RequestDestination.Companion.EMPTY: RequestDestination get() =
 ""
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.DOCUMENT: RequestDestination get() =
 "document"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.EMBED: RequestDestination get() =
 "embed"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.FONT: RequestDestination get() =
 "font"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.IMAGE: RequestDestination get() =
 "image"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.MANIFEST: RequestDestination get() =
 "manifest"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.MEDIA: RequestDestination get() =
 "media"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.OBJECT: RequestDestination get() =
 "object"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.REPORT: RequestDestination get() =
 "report"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.SCRIPT: RequestDestination get() =
 "script"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.SERVICEWORKER: RequestDestination get() =
 "serviceworker"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.SHAREDWORKER: RequestDestination get() =
 "sharedworker"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.STYLE: RequestDestination get() =
 "style"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.WORKER: RequestDestination get() =
 "worker"
.asDynamic().unsafeCast<RequestDestination>()
public inline val
RequestDestination.Companion.XSLT: RequestDestination get() =
 "xslt"
.asDynamic().unsafeCast<RequestDestination>()
}
/* please, don't implement this interface! */
public
external interface RequestMode {
 companion object
}
public inline val
RequestMode.Companion.NAVIGATE: RequestMode get() =
 "navigate"
.asDynamic().unsafeCast<RequestMode>()
public inline val
RequestMode.Companion.SAME_ORIGIN: RequestMode get() = "same-
origin"
.asDynamic().unsafeCast<RequestMode>()
public inline val
RequestMode.Companion.NO_CORS:
RequestMode get() = "no-cors"
.asDynamic().unsafeCast<RequestMode>()
public inline val
RequestMode.Companion.CORS: RequestMode get() = "cors"
.asDynamic().unsafeCast<RequestMode>()
}
/* please, don't implement this interface! */
public
external interface RequestCredentials {
 companion
object
}
public inline val
RequestCredentials.Companion.OMIT: RequestCredentials get() =
 "omit"
.asDynamic().unsafeCast<RequestCredentials>()
public inline val
RequestCredentials.Companion.SAME_ORIGIN: RequestCredentials get() = "same-
origin"
.asDynamic().unsafeCast<RequestCredentials>()
public inline val
RequestCredentials.Companion.INCLUDE: RequestCredentials get() =
 "include"
.asDynamic().unsafeCast<RequestCredentials>()
}
/* please, don't implement this interface! */
public
external interface RequestCache {
 companion object
}
public inline val
RequestCache.Companion.DEFAULT: RequestCache get() =
 "default"
.asDynamic().unsafeCast<RequestCache>()
public inline val
RequestCache.Companion.NO_STORE:

```

```

RequestCache get() = "no-store".asDynamic().unsafeCast<RequestCache>()\r\npublic inline val
RequestCache.Companion.RELOAD: RequestCache get() =
"reload".asDynamic().unsafeCast<RequestCache>()\r\npublic inline val RequestCache.Companion.NO_CACHE:
RequestCache get() = "no-cache".asDynamic().unsafeCast<RequestCache>()\r\npublic inline val
RequestCache.Companion.FORCE_CACHE: RequestCache get() = "force-
cache".asDynamic().unsafeCast<RequestCache>()\r\npublic inline val
RequestCache.Companion.ONLY_IF_CACHED: RequestCache get() = "only-if-
cached".asDynamic().unsafeCast<RequestCache>()\r\n\r\n/* please, don't implement this interface! */\r\npublic
external interface RequestRedirect {\r\n companion object\r\n}\r\npublic inline val
RequestRedirect.Companion.FOLLOW: RequestRedirect get() =
"follow".asDynamic().unsafeCast<RequestRedirect>()\r\npublic inline val RequestRedirect.Companion.ERROR:
RequestRedirect get() = "error".asDynamic().unsafeCast<RequestRedirect>()\r\npublic inline val
RequestRedirect.Companion.MANUAL: RequestRedirect get() =
"manual".asDynamic().unsafeCast<RequestRedirect>()\r\n\r\n/* please, don't implement this interface! */\r\npublic
external interface ResponseType {\r\n companion object\r\n}\r\npublic inline val
ResponseType.Companion.BASIC: ResponseType get() =
"basic".asDynamic().unsafeCast<ResponseType>()\r\npublic inline val ResponseType.Companion.CORS:
ResponseType get() = "cors".asDynamic().unsafeCast<ResponseType>()\r\npublic inline val
ResponseType.Companion.DEFAULT: ResponseType get() =
"default".asDynamic().unsafeCast<ResponseType>()\r\npublic inline val ResponseType.Companion.ERROR:
ResponseType get() = "error".asDynamic().unsafeCast<ResponseType>()\r\npublic inline val
ResponseType.Companion.OPAQUE: ResponseType get() =
"opaque".asDynamic().unsafeCast<ResponseType>()\r\npublic inline val
ResponseType.Companion.OPAQUEREDIRECT: ResponseType get() =
"opaqueredirect".asDynamic().unsafeCast<ResponseType>()\r\n\r\n"/>\r\n * Copyright 2010-2018 JetBrains
s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0
license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n// NOTE: THIS FILE IS AUTO-
GENERATED, DO NOT EDIT!\r\n// See libraries/tools/idl2k for
details\r\n\r\n@file:Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\r\npackage
org.w3c.dom.mediacapture\r\n\r\nimport kotlin.js.*\r\nimport org.khronos.webgl.*\r\nimport
org.w3c.css.masking.*\r\nimport org.w3c.dom.*\r\nimport org.w3c.dom.clipboard.*\r\nimport
org.w3c.dom.css.*\r\nimport org.w3c.dom.events.*\r\nimport org.w3c.dom.parsing.*\r\nimport
org.w3c.dom.pointerevents.*\r\nimport org.w3c.dom.svg.*\r\nimport org.w3c.dom.url.*\r\nimport
org.w3c.fetch.*\r\nimport org.w3c.files.*\r\nimport org.w3c.notifications.*\r\nimport
org.w3c.performance.*\r\nimport org.w3c.workers.*\r\nimport org.w3c.xhr.*\r\n\r\n/**\r\n * Exposes the JavaScript
[MediaStream](https://developer.mozilla.org/en/docs/Web/API/MediaStream) to Kotlin\r\n */\r\npublic external
open class MediaStream() : EventTarget {\r\n constructor(stream: MediaStream)\r\n constructor(tracks:
Array<MediaStreamTrack>)\r\n open val id: String\r\n open val active: Boolean\r\n var onaddtrack:
((MediaStreamTrackEvent) -> dynamic)?\r\n var onremovetrack: ((MediaStreamTrackEvent) -> dynamic)?\r\n
fun getAudioTracks(): Array<MediaStreamTrack>\r\n fun getVideoTracks(): Array<MediaStreamTrack>\r\n
fun getTracks(): Array<MediaStreamTrack>\r\n fun getTrackById(trackId: String): MediaStreamTrack?\r\n fun
addTrack(track: MediaStreamTrack): Unit\r\n fun removeTrack(track: MediaStreamTrack): Unit\r\n fun clone():
MediaStream\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[MediaStreamTrack](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrack) to Kotlin\r\n */\r\npublic
external abstract class MediaStreamTrack : EventTarget {\r\n open val kind: String\r\n open val id: String\r\n
open val label: String\r\n open var enabled: Boolean\r\n open val muted: Boolean\r\n open var onmute:
((Event) -> dynamic)?\r\n open var onunmute: ((Event) -> dynamic)?\r\n open val readyState:
MediaStreamTrackState\r\n open var onended: ((Event) -> dynamic)?\r\n open var onoverconstrained: ((Event) -

```

```

> dynamic)?\r\n fun clone(): MediaStreamTrack\r\n fun stop(): Unit\r\n fun getCapabilities():
MediaTrackCapabilities\r\n fun getConstraints(): MediaTrackConstraints\r\n fun getSettings():
MediaTrackSettings\r\n fun applyConstraints(constraints: MediaTrackConstraints = definedExternally):
Promise<Unit>\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[MediaTrackSupportedConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSupportedConstrai
ts) to Kotlin\r\n */\r\n\r\npublic external interface MediaTrackSupportedConstraints {\r\n var width: Boolean? /* =
true */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var height: Boolean? /* = true
/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var aspectRatio: Boolean? / = true
/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var frameRate: Boolean? / = true
/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var facingMode: Boolean? / = true
/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var resizeMode: Boolean? / = true
/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var volume: Boolean? / = true
/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var sampleRate: Boolean? / = true
/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var sampleSize: Boolean? / = true
/\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var echoCancellation: Boolean? /
= true */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var autoGainControl:
Boolean? /* = true */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var
noiseSuppression: Boolean? /* = true */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n
var latency: Boolean? /* = true */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var
channelCount: Boolean? /* = true */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n
var deviceId: Boolean? /* = true */\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var
groupId: Boolean? /* = true */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
MediaTrackSupportedConstraints(width: Boolean? = true, height: Boolean? = true, aspectRatio: Boolean? = true,
frameRate: Boolean? = true, facingMode: Boolean? = true, resizeMode: Boolean? = true, volume: Boolean? = true,
sampleRate: Boolean? = true, sampleSize: Boolean? = true, echoCancellation: Boolean? = true, autoGainControl:
Boolean? = true, noiseSuppression: Boolean? = true, latency: Boolean? = true, channelCount: Boolean? = true,
deviceId: Boolean? = true, groupId: Boolean? = true): MediaTrackSupportedConstraints {\r\n val o =
js("{}")\r\n\r\n o["width"] = width\r\n o["height"] = height\r\n o["aspectRatio"] = aspectRatio\r\n
o["frameRate"] = frameRate\r\n o["facingMode"] = facingMode\r\n o["resizeMode"] = resizeMode\r\n
o["volume"] = volume\r\n o["sampleRate"] = sampleRate\r\n o["sampleSize"] = sampleSize\r\n
o["echoCancellation"] = echoCancellation\r\n o["autoGainControl"] = autoGainControl\r\n
o["noiseSuppression"] = noiseSuppression\r\n o["latency"] = latency\r\n o["channelCount"] =
channelCount\r\n o["deviceId"] = deviceId\r\n o["groupId"] = groupId\r\n\r\n return o\r\n}\r\n\r\npublic
external interface MediaTrackCapabilities {\r\n var width: ULongRange?\r\n get() = definedExternally\r\n
set(value) = definedExternally\r\n var height: ULongRange?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var aspectRatio: DoubleRange?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var frameRate: DoubleRange?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var facingMode: Array<String>?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var resizeMode: Array<String>?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var volume: DoubleRange?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var sampleRate: ULongRange?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var sampleSize: ULongRange?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var echoCancellation: Array<Boolean>?\r\n get() = definedExternally\r\n set(value)
= definedExternally\r\n var autoGainControl: Array<Boolean>?\r\n get() = definedExternally\r\n
set(value) = definedExternally\r\n var noiseSuppression: Array<Boolean>?\r\n get() = definedExternally\r\n
set(value) = definedExternally\r\n var latency: DoubleRange?\r\n get() = definedExternally\r\n set(value)
= definedExternally\r\n var channelCount: ULongRange?\r\n get() = definedExternally\r\n set(value) =

```

```

definedExternally\r\n var deviceId: String?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var groupId: String?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun MediaTrackCapabilities(width:
ULongRange? = undefined, height: ULongRange? = undefined, aspectRatio: DoubleRange? = undefined,
frameRate: DoubleRange? = undefined, facingMode: Array<String>? = undefined, resizeMode: Array<String>? =
undefined, volume: DoubleRange? = undefined, sampleRate: ULongRange? = undefined, sampleSize:
ULongRange? = undefined, echoCancellation: Array<Boolean>? = undefined, autoGainControl: Array<Boolean>?
= undefined, noiseSuppression: Array<Boolean>? = undefined, latency: DoubleRange? = undefined, channelCount:
ULongRange? = undefined, deviceId: String? = undefined, groupId: String? = undefined): MediaTrackCapabilities
{\r\n val o = js("{}")\r\n\r\n o["width"] = width\r\n o["height"] = height\r\n o["aspectRatio"] =
aspectRatio\r\n o["frameRate"] = frameRate\r\n o["facingMode"] = facingMode\r\n o["resizeMode"] =
resizeMode\r\n o["volume"] = volume\r\n o["sampleRate"] = sampleRate\r\n o["sampleSize"] =
sampleSize\r\n o["echoCancellation"] = echoCancellation\r\n o["autoGainControl"] = autoGainControl\r\n
o["noiseSuppression"] = noiseSuppression\r\n o["latency"] = latency\r\n o["channelCount"] =
channelCount\r\n o["deviceId"] = deviceId\r\n o["groupId"] = groupId\r\n\r\n return o\r\n}\r\n\r\n/**
Exposes the JavaScript

```

```

[MediaTrackConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackConstraints) to Kotlin\r\n
*\r\n\r\npublic external interface MediaTrackConstraints : MediaTrackConstraintSet {\r\n var advanced:
Array<MediaTrackConstraintSet>?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun MediaTrackConstraints(advanced:
Array<MediaTrackConstraintSet>? = undefined, width: dynamic = undefined, height: dynamic = undefined,
aspectRatio: dynamic = undefined, frameRate: dynamic = undefined, facingMode: dynamic = undefined,
resizeMode: dynamic = undefined, volume: dynamic = undefined, sampleRate: dynamic = undefined, sampleSize:
dynamic = undefined, echoCancellation: dynamic = undefined, autoGainControl: dynamic = undefined,
noiseSuppression: dynamic = undefined, latency: dynamic = undefined, channelCount: dynamic = undefined,
deviceId: dynamic = undefined, groupId: dynamic = undefined): MediaTrackConstraints {\r\n val o =
js("{}")\r\n\r\n o["advanced"] = advanced\r\n o["width"] = width\r\n o["height"] = height\r\n
o["aspectRatio"] = aspectRatio\r\n o["frameRate"] = frameRate\r\n o["facingMode"] = facingMode\r\n
o["resizeMode"] = resizeMode\r\n o["volume"] = volume\r\n o["sampleRate"] = sampleRate\r\n
o["sampleSize"] = sampleSize\r\n o["echoCancellation"] = echoCancellation\r\n o["autoGainControl"] =
autoGainControl\r\n o["noiseSuppression"] = noiseSuppression\r\n o["latency"] = latency\r\n
o["channelCount"] = channelCount\r\n o["deviceId"] = deviceId\r\n o["groupId"] = groupId\r\n\r\n return
o\r\n}\r\n\r\n\r\npublic external interface MediaTrackConstraintSet {\r\n var width: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var height: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var aspectRatio: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var frameRate: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var facingMode: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var resizeMode: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var volume: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var sampleRate: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var sampleSize: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var echoCancellation: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var autoGainControl: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var noiseSuppression: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var latency: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var channelCount: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var deviceId: dynamic\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var groupId: dynamic\r\n get() =

```





```

MediaStreamTrackEventInit {
 val o = js("{}")
 o["track"] = track
 o["bubbles"] = bubbles
 o["cancelable"] = cancelable
 o["composed"] = composed
 return o
}

public external open class OverconstrainedErrorEvent(type: String, eventInitDict: OverconstrainedErrorEventInit) : Event {
 open val error: dynamic
}

public external interface OverconstrainedErrorEventInit : EventInit {
 var error: dynamic /* = null */
 get() = definedExternally
 set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun OverconstrainedErrorEventInit(error: dynamic = null, bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): OverconstrainedErrorEventInit {
 val o = js("{}")
 o["error"] = error
 o["bubbles"] = bubbles
 o["cancelable"] = cancelable
 o["composed"] = composed
 return o
}

/**
 * Exposes the JavaScript [MediaDevices](https://developer.mozilla.org/en/docs/Web/API/MediaDevices) to Kotlin
 */
public external abstract class MediaDevices : EventTarget {
 open var ondevicechange: ((Event) -> dynamic)?
 fun enumerateDevices(): Promise<dynamic>
 fun getSupportedConstraints(): MediaTrackSupportedConstraints
 fun getUserMedia(constraints: MediaStreamConstraints = definedExternally): Promise<MediaStream>
}

/**
 * Exposes the JavaScript [MediaDeviceInfo](https://developer.mozilla.org/en/docs/Web/API/MediaDeviceInfo) to Kotlin
 */
public external abstract class MediaDeviceInfo {
 open val deviceId: String
 open val kind: MediaDeviceKind
 open val label: String
 open val groupId: String
 fun toJSON(): dynamic
}

public external abstract class InputDeviceInfo : MediaDeviceInfo {
 fun getCapabilities(): MediaTrackCapabilities
}

/**
 * Exposes the JavaScript [MediaStreamConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaStreamConstraints) to Kotlin
 */
public external interface MediaStreamConstraints {
 var video: dynamic /* = false */
 get() = definedExternally
 set(value) = definedExternally
 var audio: dynamic /* = false */
 get() = definedExternally
 set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MediaStreamConstraints(video: dynamic = false, audio: dynamic = false): MediaStreamConstraints {
 val o = js("{}")
 o["video"] = video
 o["audio"] = audio
 return o
}

public external interface ConstrainingPattern {
 var onoverconstrained: ((Event) -> dynamic)?
 fun getCapabilities(): Capabilities
 fun getConstraints(): Constraints
 fun getSettings(): Settings
 fun applyConstraints(constraints: Constraints = definedExternally): Promise<Unit>
}

/**
 * Exposes the JavaScript [DoubleRange](https://developer.mozilla.org/en/docs/Web/API/DoubleRange) to Kotlin
 */
public external interface DoubleRange {
 var max: Double?
 get() = definedExternally
 set(value) = definedExternally
 var min: Double?
 get() = definedExternally
 set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun DoubleRange(max: Double? = undefined, min: Double? = undefined): DoubleRange {
 val o = js("{}")
 o["max"] = max
 o["min"] = min
 return o
}

public external interface ConstrainDoubleRange : DoubleRange {
 var exact: Double?
 get() = definedExternally
 set(value) = definedExternally
 var ideal: Double?
 get() = definedExternally
 set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ConstrainDoubleRange(exact: Double? = undefined, ideal: Double? = undefined, max: Double? = undefined, min: Double? = undefined): ConstrainDoubleRange {
 val o = js("{}")
 o["exact"] = exact
 o["ideal"] = ideal
 o["max"] = max
 o["min"] = min
 return o
}

public external interface ULongRange {
 var max: Int?
 get() = definedExternally
 set(value) = definedExternally
 var min: Int?
 get() = definedExternally
 set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ULongRange(max: Int? = undefined, min: Int? = undefined): ULongRange {
 val o = js("{}")
 o["max"] = max
 o["min"] = min
 return o
}

public external interface ConstrainULongRange : ULongRange {
 var exact: Int?
 get() = definedExternally
 set(value) = definedExternally
 var ideal: Int?
 get() = definedExternally
 set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ConstrainULongRange(exact: Int? = undefined, ideal: Int? = undefined, max: Int? = undefined, min: Int? = undefined): ConstrainULongRange {
 val

```





```

open class PointerEvent(type: String, eventInitDict: PointerEventInit = definedExternally) : MouseEvent {
 open val pointerId: Int
 open val width: Double
 open val height: Double
 open val pressure: Float
 open val tangentialPressure: Float
 open val tiltX: Int
 open val tiltY: Int
 open val twist: Int
 open val pointerType: String
 open val isPrimary: Boolean
}

/* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See libraries/tools/idl2k for details

@file:Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
package org.w3c.dom.svg

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/* Exposes the JavaScript [SVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGElement) to Kotlin
 */
public external abstract class SVGElement : Element, ElementCSSInlineStyle, GlobalEventHandlers, SVGElementInstance {
 open val dataset: DOMStringMap
 open val ownerSVGElement: SVGSVGElement?
 open val viewportElement: SVGElement?
 open var tabIndex: Int
 fun focus(): Unit
 fun blur(): Unit

 public external interface SVGBoundingBoxOptions {
 var fill: Boolean? /* = true */
 get() = definedExternally
 set(value) = definedExternally
 var stroke: Boolean? /* = false */
 get() = definedExternally
 set(value) = definedExternally
 var markers: Boolean? /* = false */
 get() = definedExternally
 set(value) = definedExternally
 var clipped: Boolean? /* = false */
 get() = definedExternally
 set(value) = definedExternally
 }

 @kotlin.internal.InlineOnly
 public inline fun SVGBoundingBoxOptions(fill: Boolean? = true, stroke: Boolean? = false, markers: Boolean? = false, clipped: Boolean? = false): SVGBoundingBoxOptions {
 val o = js("{}")
 o["fill"] = fill
 o["stroke"] = stroke
 o["markers"] = markers
 o["clipped"] = clipped
 return o
 }

 /* Exposes the JavaScript [SVGGraphicsElement](https://developer.mozilla.org/en/docs/Web/API/SVGGraphicsElement) to Kotlin
 */
 public external abstract class SVGGraphicsElement : SVGElement, SVGTests {
 open val transform: SVGAnimatedTransformList
 fun getBBox(options: SVGBoundingBoxOptions = definedExternally): DOMRect
 fun getCTM(): DOMMatrix?
 fun getScreenCTM(): DOMMatrix?

 }

 /* Exposes the JavaScript [SVGGeometryElement](https://developer.mozilla.org/en/docs/Web/API/SVGGeometryElement) to Kotlin
 */
 public external abstract class SVGGeometryElement : SVGGraphicsElement {
 open val pathLength: SVGAnimatedNumber
 fun isPointInFill(point: DOMPoint): Boolean
 fun isPointInStroke(point: DOMPoint): Boolean
 fun getTotalLength(): Float
 fun getPointAtLength(distance: Float): DOMPoint

 }

 /* Exposes the JavaScript [SVGNumber](https://developer.mozilla.org/en/docs/Web/API/SVGNumber) to Kotlin
 */
 public external abstract class SVGNumber {
 open var value: Float

 }

 /* Exposes the JavaScript [SVGLength](https://developer.mozilla.org/en/docs/Web/API/SVGLength) to Kotlin
 */
 public external abstract class SVGLength {
 open val unitType: Short
 open var value: Float
 open var valueInSpecifiedUnits: Float
 open var valueAsString: String
 fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float): Unit
 fun convertToSpecifiedUnits(unitType: Short): Unit

 companion object {
 val SVG_LENGTHTYPE_UNKNOWN: Short
 val SVG_LENGTHTYPE_NUMBER: Short
 val SVG_LENGTHTYPE_PERCENTAGE: Short
 val SVG_LENGTHTYPE_EMS: Short
 val SVG_LENGTHTYPE_EXS: Short
 val SVG_LENGTHTYPE_PX: Short
 val SVG_LENGTHTYPE_CM: Short
 val SVG_LENGTHTYPE_MM: Short
 val SVG_LENGTHTYPE_IN: Short
 val SVG_LENGTHTYPE_PT: Short
 val SVG_LENGTHTYPE_PC: Short
 }

 }

 /* Exposes

```

the JavaScript [SVGAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAngle) to Kotlin

```

public external abstract class SVGAngle {
 open val unitType: Short
 open var value: Float
 open var valueInSpecifiedUnits: Float
 open var valueAsString: String
 fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float): Unit
 fun convertToSpecifiedUnits(unitType: Short): Unit
}
companion object {
 val SVG_ANGLETYPE_UNKNOWN: Short
 val SVG_ANGLETYPE_UNSPECIFIED: Short
 val SVG_ANGLETYPE_DEG: Short
 val SVG_ANGLETYPE_RAD: Short
 val SVG_ANGLETYPE_GRAD: Short
}
}
public external abstract class SVGNameList {
 open val length: Int
 open val numberOfItems: Int
 fun clear(): Unit
 fun initialize(newItem: dynamic): dynamic
 fun getItem(index: Int): dynamic
 fun insertItemBefore(newItem: dynamic, index: Int): dynamic
 fun replaceItem(newItem: dynamic, index: Int): dynamic
 fun removeItem(index: Int): dynamic
 fun appendItem(newItem: dynamic): dynamic
}
@kotlin.internal.InlineOnly inline operator fun SVGNameList.get(index: Int): dynamic = asDynamic()[index]
@kotlin.internal.InlineOnly inline operator fun SVGNameList.set(index: Int, newItem: dynamic): Unit { asDynamic()[index] = newItem; }
}
Exposes the JavaScript [SVGNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGNumberList) to Kotlin
public external abstract class SVGNumberList {
 open val length: Int
 open val numberOfItems: Int
 fun clear(): Unit
 fun initialize(newItem: SVGNumber): SVGNumber
 fun getItem(index: Int): SVGNumber
 fun insertItemBefore(newItem: SVGNumber, index: Int): SVGNumber
 fun replaceItem(newItem: SVGNumber, index: Int): SVGNumber
 fun removeItem(index: Int): SVGNumber
 fun appendItem(newItem: SVGNumber): SVGNumber
}
@kotlin.internal.InlineOnly inline operator fun SVGNumberList.get(index: Int): SVGNumber? = asDynamic()[index]
@kotlin.internal.InlineOnly inline operator fun SVGNumberList.set(index: Int, newItem: SVGNumber): Unit { asDynamic()[index] = newItem; }
}
Exposes the JavaScript [SVGLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGLengthList) to Kotlin
public external abstract class SVGLengthList {
 open val length: Int
 open val numberOfItems: Int
 fun clear(): Unit
 fun initialize(newItem: SVGLength): SVGLength
 fun getItem(index: Int): SVGLength
 fun insertItemBefore(newItem: SVGLength, index: Int): SVGLength
 fun replaceItem(newItem: SVGLength, index: Int): SVGLength
 fun removeItem(index: Int): SVGLength
 fun appendItem(newItem: SVGLength): SVGLength
}
@kotlin.internal.InlineOnly inline operator fun SVGLengthList.get(index: Int): SVGLength? = asDynamic()[index]
@kotlin.internal.InlineOnly inline operator fun SVGLengthList.set(index: Int, newItem: SVGLength): Unit { asDynamic()[index] = newItem; }
}
Exposes the JavaScript [SVGAnimatedBoolean](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedBoolean) to Kotlin
public external abstract class SVGAnimatedBoolean {
 open var baseVal: Boolean
 open val animVal: Boolean
}
Exposes the JavaScript [SVGAnimatedEnumeration](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedEnumeration) to Kotlin
public external abstract class SVGAnimatedEnumeration {
 open var baseVal: Short
 open val animVal: Short
}
Exposes the JavaScript [SVGAnimatedInteger](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedInteger) to Kotlin
public external abstract class SVGAnimatedInteger {
 open var baseVal: Int
 open val animVal: Int
}
Exposes the JavaScript [SVGAnimatedNumber](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumber) to Kotlin
public external abstract class SVGAnimatedNumber {
 open var baseVal: Float
 open val animVal: Float
}
Exposes the JavaScript [SVGAnimatedLength](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLength) to Kotlin
public external abstract class SVGAnimatedLength {
 open val baseVal: SVGLength
 open val animVal: SVGLength
}
Exposes the JavaScript [SVGAnimatedAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedAngle) to Kotlin
public external abstract class SVGAnimatedAngle {
 open val baseVal: SVGAngle
 open val

```



```

suspendRedrawAll(): Unit
fun forceRedraw(): Unit
companion object {
 val
SVG_ZOOMANDPAN_UNKNOWN: Short
 val SVG_ZOOMANDPAN_DISABLE: Short
 val
SVG_ZOOMANDPAN_MAGNIFY: Short
}
Exposes the JavaScript
[SVGGElement](https://developer.mozilla.org/en/docs/Web/API/SVGGElement) to Kotlin
public external
abstract class SVGGElement : SVGGraphicsElement {
}
SVGUnknownElement : SVGGraphicsElement
Exposes the JavaScript
[SVGDefsElement](https://developer.mozilla.org/en/docs/Web/API/SVGDefsElement) to Kotlin
public
external abstract class SVGDefsElement : SVGGraphicsElement {
}
Exposes the JavaScript
[SVGDescElement](https://developer.mozilla.org/en/docs/Web/API/SVGDescElement) to Kotlin
public
external abstract class SVGDescElement : SVGElement {
}
Exposes the JavaScript
[SVGMetadataElement](https://developer.mozilla.org/en/docs/Web/API/SVGMetadataElement) to Kotlin
public
external abstract class SVGMetadataElement : SVGElement {
}
Exposes the
JavaScript
[SVGTitleElement](https://developer.mozilla.org/en/docs/Web/API/SVGTitleElement) to Kotlin
public
external abstract class SVGTitleElement : SVGElement {
}
Exposes the JavaScript
[SVGSymbolElement](https://developer.mozilla.org/en/docs/Web/API/SVGSymbolElement) to Kotlin
public
external abstract class SVGSymbolElement : SVGGraphicsElement, SVGFitToViewBox
{
}
Exposes the JavaScript
[SVGUseElement](https://developer.mozilla.org/en/docs/Web/API/SVGUseElement) to Kotlin
public
external abstract class SVGUseElement : SVGGraphicsElement, SVGURIReference {
 open val x:
SVGAnimatedLength
 open val y: SVGAnimatedLength
 open val width: SVGAnimatedLength
 open val height: SVGAnimatedLength
 open val instanceRoot: SVGElement?
 open val animatedInstanceRoot:
SVGElement?
}
public external open class SVGUseElementShadowRoot : ShadowRoot {
 override
fun getElementById(elementId: String): Element?
 override fun prepend(vararg nodes: dynamic): Unit
 override fun append(vararg nodes: dynamic): Unit
 override fun querySelector(selectors: String): Element?
 override fun querySelectorAll(selectors: String): NodeList
}
public external interface SVGElementInstance
{
 val correspondingElement: SVGElement?
 val correspondingUseElement:
SVGUseElement?
}
public external open class ShadowAnimation(source: dynamic, newTarget: dynamic)
{
 open val sourceAnimation: dynamic
}
Exposes the JavaScript
[SVGSwitchElement](https://developer.mozilla.org/en/docs/Web/API/SVGSwitchElement) to Kotlin
public
external abstract class SVGSwitchElement : SVGGraphicsElement {
}
public external
interface GetSVGDocument {
 fun getSVGDocument(): Document
}
Exposes the JavaScript
[SVGStyleElement](https://developer.mozilla.org/en/docs/Web/API/SVGStyleElement) to Kotlin
public
external abstract class SVGStyleElement : SVGElement, LinkStyle {
 open var type: String
 open var
media: String
 open var title: String
}
Exposes the JavaScript
[SVGTransform](https://developer.mozilla.org/en/docs/Web/API/SVGTransform) to Kotlin
public
external
abstract class SVGTransform {
 open val type: Short
 open val matrix: DOMMatrix
 open val angle:
Float
 fun setMatrix(matrix: DOMMatrixReadOnly): Unit
 fun setTranslate(tx: Float, ty: Float): Unit
 fun
setScale(sx: Float, sy: Float): Unit
 fun setRotate(angle: Float, cx: Float, cy: Float): Unit
 fun
setSkewX(angle: Float): Unit
 fun setSkewY(angle: Float): Unit
}
companion object {
 val
SVG_TRANSFORM_UNKNOWN: Short
 val SVG_TRANSFORM_MATRIX: Short
 val
SVG_TRANSFORM_TRANSLATE: Short
 val SVG_TRANSFORM_SCALE: Short
 val
SVG_TRANSFORM_ROTATE: Short
 val SVG_TRANSFORM_SKEWX: Short
 val
SVG_TRANSFORM_SKEWY: Short
}
Exposes the JavaScript
[SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
public
external abstract class SVGTransformList {
 open val length: Int
 open val numberOfItems: Int
 fun
clear(): Unit
 fun initialize(newItem: SVGTransform): SVGTransform
 fun getItem(index: Int):
SVGTransform
 fun insertItemBefore(newItem: SVGTransform, index: Int): SVGTransform
 fun
replaceItem(newItem: SVGTransform, index: Int): SVGTransform
 fun removeItem(index: Int):

```

```

SVGTransform fun appendItem(newItem: SVGTransform): SVGTransform fun
createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly): SVGTransform fun consolidate():
SVGTransform?@kotlin.internal.InlineOnly inline operator fun SVGTransformList.get(index: Int):
SVGTransform? = asDynamic()[index]@kotlin.internal.InlineOnly inline operator fun
SVGTransformList.set(index: Int, newItem: SVGTransform): Unit { asDynamic()[index] = newItem; }
* Exposes the JavaScript
[SVGAnimatedTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedTransformList) to
Kotlin public external abstract class SVGAnimatedTransformList { open val baseVal:
SVGTransformList open val animVal: SVGTransformList }
* Exposes the JavaScript
[SVGPreserveAspectRatio](https://developer.mozilla.org/en/docs/Web/API/SVGPreserveAspectRatio) to Kotlin
public external abstract class SVGPreserveAspectRatio { open var align: Short open var
meetOrSlice: Short companion object { val SVG_PRESERVEASPECTRATIO_UNKNOWN:
Short val SVG_PRESERVEASPECTRATIO_NONE: Short val
SVG_PRESERVEASPECTRATIO_XMINYMIN: Short val
SVG_PRESERVEASPECTRATIO_XMIDYMIN: Short val
SVG_PRESERVEASPECTRATIO_XMAXYMIN: Short val
SVG_PRESERVEASPECTRATIO_XMINYMID: Short val
SVG_PRESERVEASPECTRATIO_XMIDYMID: Short val
SVG_PRESERVEASPECTRATIO_XMAXYMID: Short val
SVG_PRESERVEASPECTRATIO_XMINYMAX: Short val
SVG_PRESERVEASPECTRATIO_XMIDYMAX: Short val
SVG_PRESERVEASPECTRATIO_XMAXYMAX: Short val SVG_MEETORSLICE_UNKNOWN:
Short val SVG_MEETORSLICE_MEET: Short val SVG_MEETORSLICE_SLICE: Short
} }
* Exposes the JavaScript
[SVGAnimatedPreserveAspectRatio](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPreserveAspect
Ratio) to Kotlin public external abstract class SVGAnimatedPreserveAspectRatio { open val baseVal:
SVGPreserveAspectRatio open val animVal: SVGPreserveAspectRatio }
* Exposes the
JavaScript [SVGPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGPathElement) to Kotlin
public external abstract class SVGPathElement : SVGGeometryElement { }
* Exposes the
JavaScript [SVGRectElement](https://developer.mozilla.org/en/docs/Web/API/SVGRectElement) to Kotlin
public external abstract class SVGRectElement : SVGGeometryElement { open val x:
SVGAnimatedLength open val y: SVGAnimatedLength open val width: SVGAnimatedLength open
val height: SVGAnimatedLength open val rx: SVGAnimatedLength open val ry:
SVGAnimatedLength }
* Exposes the JavaScript
[SVGCircleElement](https://developer.mozilla.org/en/docs/Web/API/SVGCircleElement) to Kotlin
public external abstract class SVGCircleElement : SVGGeometryElement { open val cx: SVGAnimatedLength
open val cy: SVGAnimatedLength open val r: SVGAnimatedLength }
* Exposes the
JavaScript [SVGEllipseElement](https://developer.mozilla.org/en/docs/Web/API/SVGEllipseElement) to Kotlin
public external abstract class SVGEllipseElement : SVGGeometryElement { open val cx:
SVGAnimatedLength open val cy: SVGAnimatedLength open val rx: SVGAnimatedLength open
val ry: SVGAnimatedLength }
* Exposes the JavaScript
[SVGLineElement](https://developer.mozilla.org/en/docs/Web/API/SVGLineElement) to Kotlin
public external abstract class SVGLineElement : SVGGeometryElement { open val x1: SVGAnimatedLength
open val y1: SVGAnimatedLength open val x2: SVGAnimatedLength open val y2:
SVGAnimatedLength }
* Exposes the JavaScript
[SVGMeshElement](https://developer.mozilla.org/en/docs/Web/API/SVGMeshElement) to Kotlin
public external abstract class SVGMeshElement : SVGGeometryElement, SVGURIReference { }
* Exposes the JavaScript

```



[SVGAnimatedPoints](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPoints) to Kotlin\r\n
 \*\\r\npublic external interface SVGAnimatedPoints {\\r\n val points: SVGPointList\\r\n val animatedPoints: SVGPointList\\r\n}\\r\n\\r\npublic external abstract class SVGPointList {\\r\n open val length: Int\\r\n open val numberOfItems: Int\\r\n fun clear(): Unit\\r\n fun initialize(newItem: DOMPoint): DOMPoint\\r\n fun getItem(index: Int): DOMPoint\\r\n fun insertItemBefore(newItem: DOMPoint, index: Int): DOMPoint\\r\n fun replaceItem(newItem: DOMPoint, index: Int): DOMPoint\\r\n fun removeItem(index: Int): DOMPoint\\r\n fun appendItem(newItem: DOMPoint): DOMPoint\\r\n}\\r\n@kotlin.internal.InlineOnly inline operator fun SVGPointList.get(index: Int): DOMPoint? = asDynamic()[index]\\r\n@kotlin.internal.InlineOnly inline operator fun SVGPointList.set(index: Int, newItem: DOMPoint): Unit { asDynamic()[index] = newItem; }\\r\n\\r\n\*\*\\r\n \* Exposes the JavaScript

[SVGPolylineElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolylineElement) to Kotlin\r\n
 \*\\r\npublic external abstract class SVGPolylineElement : SVGGeometryElement, SVGAnimatedPoints {\\r\n}\\r\n\\r\n\\r\n\*\*\\r\n \* Exposes the JavaScript

[SVGPolygonElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolygonElement) to Kotlin\r\n
 \*\\r\npublic external abstract class SVGPolygonElement : SVGGeometryElement, SVGAnimatedPoints {\\r\n}\\r\n\\r\n\\r\n\*\*\\r\n \* Exposes the JavaScript

[SVGTextContentElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextContentElement) to Kotlin\r\n
 \*\\r\npublic external abstract class SVGTextContentElement : SVGGraphicsElement {\\r\n open val textLength: SVGAnimatedLength\\r\n open val lengthAdjust: SVGAnimatedEnumeration\\r\n fun getNumberOfChars(): Int\\r\n fun getComputedTextLength(): Float\\r\n fun getSubStringLength(charnum: Int, nchars: Int): Float\\r\n fun getStartPositionOfChar(charnum: Int): DOMPoint\\r\n fun getEndPositionOfChar(charnum: Int): DOMPoint\\r\n fun getExtentOfChar(charnum: Int): DOMRect\\r\n fun getRotationOfChar(charnum: Int): Float\\r\n fun getCharNumAtPosition(point: DOMPoint): Int\\r\n fun selectSubString(charnum: Int, nchars: Int): Unit\\r\n\\r\n companion object {\\r\n val LENGTHADJUST\_UNKNOWN: Short\\r\n val LENGTHADJUST\_SPACING: Short\\r\n val LENGTHADJUST\_SPACINGANDGLYPHS: Short\\r\n }\\r\n}\\r\n\\r\n\\r\n\*\*\\r\n \* Exposes the JavaScript

[SVGTextPositioningElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextPositioningElement) to Kotlin\r\n
 \*\\r\npublic external abstract class SVGTextPositioningElement : SVGTextContentElement {\\r\n open val x: SVGAnimatedLengthList\\r\n open val y: SVGAnimatedLengthList\\r\n open val dx: SVGAnimatedLengthList\\r\n open val dy: SVGAnimatedLengthList\\r\n open val rotate: SVGAnimatedNumberList\\r\n}\\r\n\\r\n\\r\n\*\*\\r\n \* Exposes the JavaScript

[SVGTextElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextElement) to Kotlin\r\n
 \*\\r\npublic external abstract class SVGTextElement : SVGTextPositioningElement {\\r\n}\\r\n\\r\n\\r\n\*\*\\r\n \* Exposes the JavaScript [SVGTSpanElement](https://developer.mozilla.org/en/docs/Web/API/SVGTSpanElement) to Kotlin\r\n
 \*\\r\npublic external abstract class SVGTSpanElement : SVGTextPositioningElement {\\r\n}\\r\n\\r\n\\r\n\*\*\\r\n \* Exposes the JavaScript [SVGTextPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextPathElement) to Kotlin\r\n
 \*\\r\npublic external abstract class SVGTextPathElement : SVGTextContentElement, SVGURIReference {\\r\n open val startOffset: SVGAnimatedLength\\r\n open val method: SVGAnimatedEnumeration\\r\n open val spacing: SVGAnimatedEnumeration\\r\n\\r\n companion object {\\r\n val TEXTPATH\_METHODTYPE\_UNKNOWN: Short\\r\n val TEXTPATH\_METHODTYPE\_ALIGN: Short\\r\n val TEXTPATH\_METHODTYPE\_STRETCH: Short\\r\n val TEXTPATH\_SPACINGTYPE\_UNKNOWN: Short\\r\n val TEXTPATH\_SPACINGTYPE\_AUTO: Short\\r\n val TEXTPATH\_SPACINGTYPE\_EXACT: Short\\r\n }\\r\n}\\r\n\\r\n\\r\n\*\*\\r\n \* Exposes the JavaScript

[SVGImageElement](https://developer.mozilla.org/en/docs/Web/API/SVGImageElement) to Kotlin\r\n
 \*\\r\npublic external abstract class SVGImageElement : SVGGraphicsElement, SVGURIReference, HTMLOrSVGImageElement {\\r\n open val x: SVGAnimatedLength\\r\n open val y: SVGAnimatedLength\\r\n open val width: SVGAnimatedLength\\r\n open val height: SVGAnimatedLength\\r\n open val preserveAspectRatio: SVGAnimatedPreserveAspectRatio\\r\n open var crossOrigin: String?\\r\n}\\r\n\\r\n\\r\n\*\*\\r\n \* Exposes the JavaScript

Exposes the JavaScript

[SVGForeignObjectElement](<https://developer.mozilla.org/en/docs/Web/API/SVGForeignObjectElement>) to Kotlin  
\*/  
public external abstract class SVGForeignObjectElement : SVGGraphicsElement {  
 open val x: SVGAnimatedLength  
 open val y: SVGAnimatedLength  
 open val width: SVGAnimatedLength  
 open val height: SVGAnimatedLength  
}  
public external abstract class SVGMarkerElement : SVGElement, SVGFitToViewBox {  
 open val refX: SVGAnimatedLength  
 open val refY: SVGAnimatedLength  
 open val markerUnits: SVGAnimatedEnumeration  
 open val markerWidth: SVGAnimatedLength  
 open val markerHeight: SVGAnimatedLength  
 open val orientType: SVGAnimatedEnumeration  
 open val orientAngle: SVGAnimatedAngle  
 open var orient: String  
 fun setOrientToAuto(): Unit  
 fun setOrientToAngle(angle: SVGAngle): Unit  
 companion object {  
 val  
 SVG\_MARKERUNITS\_UNKNOWN: Short  
 val SVG\_MARKERUNITS\_USERSPACEONUSE: Short  
 val SVG\_MARKERUNITS\_STROKEWIDTH: Short  
 val SVG\_MARKER\_ORIENT\_UNKNOWN: Short  
 val SVG\_MARKER\_ORIENT\_AUTO: Short  
 val SVG\_MARKER\_ORIENT\_ANGLE: Short  
 }  
}  
\*/  
Exposes the JavaScript

[SVGSolidcolorElement](<https://developer.mozilla.org/en/docs/Web/API/SVGSolidcolorElement>) to Kotlin  
\*/  
public external abstract class SVGSolidcolorElement : SVGElement {  
}  
\*/  
Exposes the JavaScript  
[SVGGradientElement](<https://developer.mozilla.org/en/docs/Web/API/SVGGradientElement>) to Kotlin  
\*/  
public external abstract class SVGGradientElement : SVGElement, SVGURIReference, SVGUnitTypes {  
 open val gradientUnits: SVGAnimatedEnumeration  
 open val gradientTransform: SVGAnimatedTransformList  
 open val spreadMethod: SVGAnimatedEnumeration  
 companion object {  
 val SVG\_SPREADMETHOD\_UNKNOWN: Short  
 val SVG\_SPREADMETHOD\_PAD: Short  
 val SVG\_SPREADMETHOD\_REFLECT: Short  
 val SVG\_SPREADMETHOD\_REPEAT: Short  
 val SVG\_UNIT\_TYPE\_UNKNOWN: Short  
 val SVG\_UNIT\_TYPE\_USERSPACEONUSE: Short  
 val SVG\_UNIT\_TYPE\_OBJECTBOUNDINGBOX: Short  
 }  
}  
\*/  
Exposes the JavaScript

[SVGLinearGradientElement](<https://developer.mozilla.org/en/docs/Web/API/SVGLinearGradientElement>) to Kotlin  
\*/  
public external abstract class SVGLinearGradientElement : SVGGradientElement {  
 open val x1: SVGAnimatedLength  
 open val y1: SVGAnimatedLength  
 open val x2: SVGAnimatedLength  
 open val y2: SVGAnimatedLength  
}  
\*/  
Exposes the JavaScript

[SVGRadialGradientElement](<https://developer.mozilla.org/en/docs/Web/API/SVGRadialGradientElement>) to Kotlin  
\*/  
public external abstract class SVGRadialGradientElement : SVGGradientElement {  
 open val cx: SVGAnimatedLength  
 open val cy: SVGAnimatedLength  
 open val r: SVGAnimatedLength  
 open val fx: SVGAnimatedLength  
 open val fy: SVGAnimatedLength  
}  
public external abstract class SVGMeshGradientElement : SVGGradientElement  
public external abstract class SVGMeshrowElement : SVGElement  
public external abstract class SVGMeshpatchElement : SVGElement {  
}  
\*/  
Exposes the JavaScript

[SVGStopElement](<https://developer.mozilla.org/en/docs/Web/API/SVGStopElement>) to Kotlin  
\*/  
public external abstract class SVGStopElement : SVGElement {  
 open val offset: SVGAnimatedNumber  
}  
\*/  
Exposes the JavaScript

[SVGPatternElement](<https://developer.mozilla.org/en/docs/Web/API/SVGPatternElement>) to Kotlin  
\*/  
public external abstract class SVGPatternElement : SVGElement, SVGFitToViewBox, SVGURIReference, SVGUnitTypes {  
 open val patternUnits: SVGAnimatedEnumeration  
 open val patternContentUnits: SVGAnimatedEnumeration  
 open val patternTransform: SVGAnimatedTransformList  
 open val x: SVGAnimatedLength  
 open val y: SVGAnimatedLength  
 open val width: SVGAnimatedLength  
 open val height: SVGAnimatedLength  
 companion object {  
 val SVG\_UNIT\_TYPE\_UNKNOWN: Short  
 val SVG\_UNIT\_TYPE\_USERSPACEONUSE: Short  
 val SVG\_UNIT\_TYPE\_OBJECTBOUNDINGBOX: Short  
 }  
}  
public external abstract class SVGHatchElement : SVGElement  
public external abstract class SVGHatchpathElement : SVGElement {  
}  
\*/  
Exposes the JavaScript

[SVGCursorElement](https://developer.mozilla.org/en/docs/Web/API/SVGCursorElement) to Kotlin

```

*public external abstract class SVGCursorElement : SVGElement, SVGURIReference {
 open val x: SVGAnimatedLength
 open val y: SVGAnimatedLength
}

```

\* Exposes the JavaScript

[SVGScriptElement](https://developer.mozilla.org/en/docs/Web/API/SVGScriptElement) to Kotlin

```

*public external abstract class SVGScriptElement : SVGElement, SVGURIReference, HTMLScriptElement {
 open var type: String
 open var crossOrigin: String?
}

```

\* Exposes the JavaScript

[SVGAEElement](https://developer.mozilla.org/en/docs/Web/API/SVGAEElement) to Kotlin

```

*public external abstract class SVGAEElement : SVGGraphicsElement, SVGURIReference {
 open val target: SVGAnimatedString
 open val download: SVGAnimatedString
 open val rel: SVGAnimatedString
 open val relList: SVGAnimatedString
 open val hreflang: SVGAnimatedString
}

```

\* Exposes the JavaScript

[SVGViewElement](https://developer.mozilla.org/en/docs/Web/API/SVGViewElement) to Kotlin

```

*public external abstract class SVGViewElement : SVGElement, SVGFitToViewBox, SVGZoomAndPan {
 companion object {
 val SVG_ZOOMANDPAN_UNKNOWN: Short
 val SVG_ZOOMANDPAN_DISABLE: Short
 val SVG_ZOOMANDPAN_MAGNIFY: Short
 }
}

```

\* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.  
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

\* NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT! See libraries/tools/idl2k for details

```

@file:Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
package org.w3c.files
import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

```

\* Exposes the JavaScript

[Blob](https://developer.mozilla.org/en/docs/Web/API/Blob) to Kotlin

```

*public external open class Blob(blobParts: Array<dynamic> = definedExternally, options: BlobPropertyBag = definedExternally) {
 open val size: Int
 open val type: String
 open val isClosed: Boolean
 fun slice(start: Int = definedExternally, end: Int = definedExternally, contentType: String = definedExternally): Blob
 fun close(): Unit
}

```

\*public external interface BlobPropertyBag {
 var type: String? /\* = "" \*/
 get() = definedExternally
 set(value) = definedExternally
}

```

@kotlin.internal.InlineOnly
public inline fun BlobPropertyBag(type: String? = ""): BlobPropertyBag {
 val o = js("{}")
 o["type"] = type
 return o
}

```

\* Exposes the JavaScript

[File](https://developer.mozilla.org/en/docs/Web/API/File) to Kotlin

```

*public external open class File(fileBits: Array<dynamic>, fileName: String, options: FilePropertyBag = definedExternally) : Blob {
 open val name: String
 open val lastModified: Int
}

```

\*public external interface FilePropertyBag : BlobPropertyBag {
 var lastModified: Int?
 get() = definedExternally
 set(value) = definedExternally
}

```

@kotlin.internal.InlineOnly
public inline fun FilePropertyBag(lastModified: Int? = undefined, type: String? = ""): FilePropertyBag {
 val o = js("{}")
 o["lastModified"] = lastModified
 o["type"] = type
 return o
}

```

\* Exposes the JavaScript

[FileList](https://developer.mozilla.org/en/docs/Web/API/FileList) to Kotlin

```

*public external abstract class FileList : ItemArrayLike<File> {
 override val length: Int
 override fun item(index: Int): File?
}

```

```

@kotlin.internal.InlineOnly
inline operator fun FileList.get(index: Int): File? = asDynamic()[index]

```

\* Exposes the JavaScript

[FileReader](https://developer.mozilla.org/en/docs/Web/API/FileReader) to Kotlin

```

*public external open class FileReader : EventTarget {
 open val readyState: Short
 open val result: dynamic
 open val error: dynamic
 var onloadstart: ((ProgressEvent) -> dynamic)?
 var onprogress: ((ProgressEvent) -> dynamic)?
 var onload: ((Event) -> dynamic)?
 var onabort: ((Event) -> dynamic)?
 var onerror:

```

```

((Event) -> dynamic)?\r\n var onloadend: ((Event) -> dynamic)?\r\n fun readAsArrayBuffer(blob: Blob):
Unit\r\n fun readAsBinaryString(blob: Blob): Unit\r\n fun readAsText(blob: Blob, label: String =
definedExternally): Unit\r\n fun readAsDataURL(blob: Blob): Unit\r\n\r\n companion
object {\r\n val EMPTY: Short\r\n val LOADING: Short\r\n val DONE: Short\r\n }\r\n}\r\n\r\n/**\r\n
* Exposes the JavaScript [FileReaderSync](https://developer.mozilla.org/en/docs/Web/API/FileReaderSync) to
Kotlin\r\n */\r\npublic external open class FileReaderSync {\r\n fun readAsArrayBuffer(blob: Blob):
ArrayBuffer\r\n fun readAsBinaryString(blob: Blob): String\r\n fun readAsText(blob: Blob, label: String =
definedExternally): String\r\n fun readAsDataURL(blob: Blob): String\r\n}\r\n\r\n", "*/\r\n * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n// NOTE: THIS FILE IS AUTO-
GENERATED, DO NOT EDIT!\r\n// See libraries/tools/idl2k for
details\r\n\r\n@file:Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\r\n\r\npackage
org.w3c.notifications\r\n\r\nimport kotlin.js.*\r\nimport org.khronos.webgl.*\r\nimport
org.w3c.css.masking.*\r\nimport org.w3c.dom.*\r\nimport org.w3c.dom.clipboard.*\r\nimport
org.w3c.dom.css.*\r\nimport org.w3c.dom.events.*\r\nimport org.w3c.dom.mediacapture.*\r\nimport
org.w3c.dom.parsing.*\r\nimport org.w3c.dom.pointerevents.*\r\nimport org.w3c.dom.svg.*\r\nimport
org.w3c.dom.url.*\r\nimport org.w3c.fetch.*\r\nimport org.w3c.files.*\r\nimport org.w3c.performance.*\r\nimport
org.w3c.workers.*\r\nimport org.w3c.xhr.*\r\n\r\n\r\n/**\r\n * Exposes the JavaScript
[Notification](https://developer.mozilla.org/en/docs/Web/API/Notification) to Kotlin\r\n */\r\npublic external open
class Notification(title: String, options: NotificationOptions = definedExternally) : EventTarget {\r\n var onclick:
((MouseEvent) -> dynamic)?\r\n var onerror: ((Event) -> dynamic)?\r\n open val title: String\r\n open val dir:
NotificationDirection\r\n open val lang: String\r\n open val body: String\r\n open val tag: String\r\n open val
image: String\r\n open val icon: String\r\n open val badge: String\r\n open val sound: String\r\n open val
vibrate: Array<out Int>\r\n open val timestamp: Number\r\n open val renotify: Boolean\r\n open val silent:
Boolean\r\n open val noscreen: Boolean\r\n open val requireInteraction: Boolean\r\n open val sticky:
Boolean\r\n open val data: Any?\r\n open val actions: Array<out NotificationAction>\r\n fun close():
Unit\r\n\r\n companion object {\r\n var permission: NotificationPermission\r\n var maxActions: Int\r\n fun requestPermission(deprecatedCallback: (NotificationPermission) -> Unit = definedExternally):
Promise<NotificationPermission>\r\n }\r\n}\r\n\r\n\r\npublic external interface NotificationOptions {\r\n var dir:
NotificationDirection? /* = NotificationDirection.AUTO */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var lang: String? /* = "" */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var body: String? /* = "" */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var tag: String? /* = "" */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var image: String?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var icon: String?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n
 var badge: String?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var sound:
String?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var vibrate: dynamic\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n var timestamp: Number?\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var renotify: Boolean? /* = false */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var silent: Boolean? /* = false */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var noscreen: Boolean? /* = false */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var requireInteraction: Boolean? /* = false */\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n var sticky: Boolean? /* = false */\r\n get()
= definedExternally\r\n set(value) = definedExternally\r\n var data: Any? /* = null */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var actions: Array<NotificationAction>? /* =
arrayOf() */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun NotificationOptions(dir:
NotificationDirection? = NotificationDirection.AUTO, lang: String? = "", body: String? = "", tag: String? = "",

```

```

image: String? = undefined, icon: String? = undefined, badge: String? = undefined, sound: String? = undefined,
vibrate: dynamic = undefined, timestamp: Number? = undefined, renotify: Boolean? = false, silent: Boolean? =
false, noscreen: Boolean? = false, requireInteraction: Boolean? = false, sticky: Boolean? = false, data: Any? = null,
actions: Array<NotificationAction>? = arrayOf(): NotificationOptions { \r\n val o = js("{}")\r\n\r\n o["dir"]
= dir\r\n o["lang"] = lang\r\n o["body"] = body\r\n o["tag"] = tag\r\n o["image"] = image\r\n
o["icon"] = icon\r\n o["badge"] = badge\r\n o["sound"] = sound\r\n o["vibrate"] = vibrate\r\n
o["timestamp"] = timestamp\r\n o["renotify"] = renotify\r\n o["silent"] = silent\r\n o["noscreen"] =
noscreen\r\n o["requireInteraction"] = requireInteraction\r\n o["sticky"] = sticky\r\n o["data"] = data\r\n
o["actions"] = actions\r\n\r\n return o\r\n}\r\n\r\npublic external interface NotificationAction { \r\n var action:
String?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var title: String?\r\n get()
= definedExternally\r\n set(value) = definedExternally\r\n var icon: String?\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
NotificationAction(action: String?, title: String?, icon: String? = undefined): NotificationAction { \r\n val o =
js("{}")\r\n\r\n o["action"] = action\r\n o["title"] = title\r\n o["icon"] = icon\r\n\r\n return
o\r\n}\r\n\r\npublic external interface GetNotificationOptions { \r\n var tag: String? /* = "" */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
GetNotificationOptions(tag: String? = ""): GetNotificationOptions { \r\n val o = js("{}")\r\n\r\n o["tag"] =
tag\r\n\r\n return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[NotificationEvent](https://developer.mozilla.org/en/docs/Web/API/NotificationEvent) to Kotlin\r\n * \r\n\r\npublic
external open class NotificationEvent(type: String, eventInitDict: NotificationEventInit) : ExtendableEvent { \r\n
open val notification: Notification\r\n open val action: String\r\n}\r\n\r\npublic external interface
NotificationEventInit : ExtendableEventInit { \r\n var notification: Notification?\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var action: String? /* = "" */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
NotificationEventInit(notification: Notification?, action: String? = "", bubbles: Boolean? = false, cancelable:
Boolean? = false, composed: Boolean? = false): NotificationEventInit { \r\n val o = js("{}")\r\n\r\n
o["notification"] = notification\r\n o["action"] = action\r\n o["bubbles"] = bubbles\r\n o["cancelable"] =
cancelable\r\n o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\n/* please, don't implement this interface!
*\r\n\r\npublic external interface NotificationPermission { \r\n companion object\r\n}\r\n\r\npublic inline val
NotificationPermission.Companion.DEFAULT: NotificationPermission get() =
"default".asDynamic().unsafeCast<NotificationPermission>()\r\n\r\npublic inline val
NotificationPermission.Companion.DENIED: NotificationPermission get() =
"denied".asDynamic().unsafeCast<NotificationPermission>()\r\n\r\npublic inline val
NotificationPermission.Companion.GRANTED: NotificationPermission get() =
"granted".asDynamic().unsafeCast<NotificationPermission>()\r\n\r\n/* please, don't implement this interface!
*\r\n\r\npublic external interface NotificationDirection { \r\n companion object\r\n}\r\n\r\npublic inline val
NotificationDirection.Companion.AUTO: NotificationDirection get() =
"auto".asDynamic().unsafeCast<NotificationDirection>()\r\n\r\npublic inline val
NotificationDirection.Companion.LTR: NotificationDirection get() =
"ltr".asDynamic().unsafeCast<NotificationDirection>()\r\n\r\npublic inline val
NotificationDirection.Companion.RTL: NotificationDirection get() =
"rtl".asDynamic().unsafeCast<NotificationDirection>()\r\n\r\n", /*\r\n * Copyright 2010-2018 JetBrains s.r.o. and
Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that
can be found in the license/LICENSE.txt file.\r\n * \r\n\r\n\r\n// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT
EDIT!\r\n\r\n// See libraries/tools/idl2k for
details\r\n\r\n@file:Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\r\n\r\npackage
org.w3c.workers\r\n\r\nimport kotlin.js.*\r\nimport org.khronos.webgl.*\r\nimport org.w3c.css.masking.*\r\nimport
org.w3c.dom.*\r\nimport org.w3c.dom.clipboard.*\r\nimport org.w3c.dom.css.*\r\nimport

```

```

org.w3c.dom.events.*\r\nimport org.w3c.dom.mediacapture.*\r\nimport org.w3c.dom.parsing.*\r\nimport
org.w3c.dom.pointerevents.*\r\nimport org.w3c.dom.svg.*\r\nimport org.w3c.dom.url.*\r\nimport
org.w3c.fetch.*\r\nimport org.w3c.files.*\r\nimport org.w3c.notifications.*\r\nimport
org.w3c.performance.*\r\nimport org.w3c.xhr.*\r\n\r\n/**\r\n * Exposes the JavaScript
[ServiceWorkerRegistration](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerRegistration) to
Kotlin\r\n */\r\npublic external abstract class ServiceWorkerRegistration : EventTarget {\r\n open val installing:
ServiceWorker?\r\n open val waiting: ServiceWorker?\r\n open val active: ServiceWorker?\r\n open val scope:
String\r\n open var onupdatefound: ((Event) -> dynamic)?\r\n open val APISpace: dynamic\r\n fun update():
Promise<Unit>\r\n fun unregister(): Promise<Boolean>\r\n fun methodName(): Promise<dynamic>\r\n fun
showNotification(title: String, options: NotificationOptions = definedExternally): Promise<Unit>\r\n fun
getNotifications(filter: GetNotificationOptions = definedExternally): Promise<dynamic>\r\n}\r\n\r\n/**\r\n *
Exposes the JavaScript
[ServiceWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerGlobalScope) to
Kotlin\r\n */\r\npublic external abstract class ServiceWorkerGlobalScope : WorkerGlobalScope {\r\n open val
clients: Clients\r\n open val registration: ServiceWorkerRegistration\r\n open var oninstall: ((Event) ->
dynamic)?\r\n open var onactivate: ((Event) -> dynamic)?\r\n open var onfetch: ((FetchEvent) -> dynamic)?\r\n
open var onforeignfetch: ((Event) -> dynamic)?\r\n open var onmessage: ((MessageEvent) -> dynamic)?\r\n
open var onfunctionalevent: ((Event) -> dynamic)?\r\n open var onnotificationclick: ((NotificationEvent) ->
dynamic)?\r\n open var onnotificationclose: ((NotificationEvent) -> dynamic)?\r\n fun skipWaiting():
Promise<Unit>\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[ServiceWorker](https://developer.mozilla.org/en/docs/Web/API/ServiceWorker) to Kotlin\r\n */\r\npublic external
abstract class ServiceWorker : EventTarget, AbstractWorker, UnionMessagePortOrServiceWorker,
UnionClientOrMessagePortOrServiceWorker {\r\n open val scriptURL: String\r\n open val state:
ServiceWorkerState\r\n open var onstatechange: ((Event) -> dynamic)?\r\n fun postMessage(message: Any?,
transfer: Array<dynamic> = definedExternally): Unit\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[ServiceWorkerContainer](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerContainer) to Kotlin\r\n
*/\r\npublic external abstract class ServiceWorkerContainer : EventTarget {\r\n open val controller:
ServiceWorker?\r\n open val ready: Promise<ServiceWorkerRegistration>\r\n open var oncontrollerchange:
((Event) -> dynamic)?\r\n open var onmessage: ((MessageEvent) -> dynamic)?\r\n fun register(scriptURL:
String, options: RegistrationOptions = definedExternally): Promise<ServiceWorkerRegistration>\r\n fun
getRegistration(clientURL: String = definedExternally): Promise<Any?>\r\n fun getRegistrations():
Promise<dynamic>\r\n fun startMessages(): Unit\r\n}\r\n\r\npublic external interface RegistrationOptions {\r\n
var scope: String?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var type:
WorkerType? /* = WorkerType.CLASSIC */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun RegistrationOptions(scope: String? =
undefined, type: WorkerType? = WorkerType.CLASSIC): RegistrationOptions {\r\n val o = js("{}")\r\n o["scope"] = scope\r\n o["type"] = type\r\n return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[ServiceWorkerMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerMessageEvent) to
Kotlin\r\n */\r\npublic external open class ServiceWorkerMessageEvent(type: String, eventInitDict:
ServiceWorkerMessageEventInit = definedExternally) : Event {\r\n open val data: Any?\r\n open val origin:
String\r\n open val lastEventId: String\r\n open val source: UnionMessagePortOrServiceWorker?\r\n open val
ports: Array<out MessagePort>?\r\n}\r\n\r\npublic external interface ServiceWorkerMessageEventInit : EventInit
{\r\n var data: Any?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var origin:
String?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var lastEventId: String?\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n var source:
UnionMessagePortOrServiceWorker?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n
var ports: Array<MessagePort>?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ServiceWorkerMessageEventInit(data:

```



```

bubbles\r\n o["cancelable"] = cancelable\r\n o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\npublic
external open class ForeignFetchEvent(type: String, eventInitDict: ForeignFetchEventInit) : ExtendableEvent {\r\n
open val request: Request\r\n open val origin: String\r\n fun respondWith(r: Promise<ForeignFetchResponse>):
Unit\r\n}\r\n\r\npublic external interface ForeignFetchEventInit : ExtendableEventInit {\r\n var request:
Request?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var origin: String? /* =
>null */\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ForeignFetchEventInit(request:
Request?, origin: String? = "null", bubbles: Boolean? = false, cancelable: Boolean? = false, composed: Boolean? =
false): ForeignFetchEventInit {\r\n val o = js("{}")\r\n o["request"] = request\r\n o["origin"] =
origin\r\n o["bubbles"] = bubbles\r\n o["cancelable"] = cancelable\r\n o["composed"] = composed\r\n\r\n
return o\r\n}\r\n\r\npublic external interface ForeignFetchResponse {\r\n var response: Response?\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var origin: String?\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var headers: Array<String>?\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
ForeignFetchResponse(response: Response?, origin: String? = undefined, headers: Array<String>? = undefined):
ForeignFetchResponse {\r\n val o = js("{}")\r\n o["response"] = response\r\n o["origin"] = origin\r\n
o["headers"] = headers\r\n\r\n return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[ExtendableMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ExtendableMessageEvent) to Kotlin\r\n
*\r\n\r\npublic external open class ExtendableMessageEvent(type: String, eventInitDict: ExtendableMessageEventInit
= definedExternally) : ExtendableEvent {\r\n open val data: Any?\r\n open val origin: String\r\n open val
lastEventId: String\r\n open val source: UnionClientOrMessagePortOrServiceWorker?\r\n open val ports:
Array<out MessagePort>?\r\n}\r\n\r\npublic external interface ExtendableMessageEventInit : ExtendableEventInit
{\r\n var data: Any?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var origin:
String?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var lastEventId: String?\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n var source:
UnionClientOrMessagePortOrServiceWorker?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n var ports: Array<MessagePort>?\r\n get() = definedExternally\r\n set(value) =
definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ExtendableMessageEventInit(data:
Any? = undefined, origin: String? = undefined, lastEventId: String? = undefined, source:
UnionClientOrMessagePortOrServiceWorker? = undefined, ports: Array<MessagePort>? = undefined, bubbles:
Boolean? = false, cancelable: Boolean? = false, composed: Boolean? = false): ExtendableMessageEventInit {\r\n
val o = js("{}")\r\n o["data"] = data\r\n o["origin"] = origin\r\n o["lastEventId"] = lastEventId\r\n
o["source"] = source\r\n o["ports"] = ports\r\n o["bubbles"] = bubbles\r\n o["cancelable"] =
cancelable\r\n o["composed"] = composed\r\n\r\n return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[Cache](https://developer.mozilla.org/en/docs/Web/API/Cache) to Kotlin\r\n
*\r\n\r\npublic external abstract class
Cache {\r\n fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>\r\n
fun matchAll(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally):
Promise<dynamic>\r\n fun add(request: dynamic): Promise<Unit>\r\n fun addAll(requests: Array<dynamic>):
Promise<Unit>\r\n fun put(request: dynamic, response: Response): Promise<Unit>\r\n fun delete(request:
dynamic, options: CacheQueryOptions = definedExternally): Promise<Boolean>\r\n fun keys(request: dynamic =
definedExternally, options: CacheQueryOptions = definedExternally): Promise<dynamic>\r\n}\r\n\r\n\r\npublic
external interface CacheQueryOptions {\r\n var ignoreSearch: Boolean? /* = false */\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n var ignoreMethod: Boolean? /* = false */\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n var ignoreVary: Boolean? /* = false */\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n var cacheName: String?\r\n get() =
definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
CacheQueryOptions(ignoreSearch: Boolean? = false, ignoreMethod: Boolean? = false, ignoreVary: Boolean? =
false, cacheName: String? = undefined): CacheQueryOptions {\r\n val o = js("{}")\r\n o["ignoreSearch"]

```



```

= ignoreSearch\r\n o["ignoreMethod"] = ignoreMethod\r\n o["ignoreVary"] = ignoreVary\r\n
o["cacheName"] = cacheName\r\n\r\n return o\r\n}\r\n\r\npublic external interface CacheBatchOperation {\r\n
var type: String?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var request:
Request?\r\n get() = definedExternally\r\n set(value) = definedExternally\r\n var response: Response?\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n var options: CacheQueryOptions?\r\n
get() = definedExternally\r\n set(value) = definedExternally\r\n}\r\n\r\n@kotlin.internal.InlineOnly\r\npublic
inline fun CacheBatchOperation(type: String? = undefined, request: Request? = undefined, response: Response? =
undefined, options: CacheQueryOptions? = undefined): CacheBatchOperation {\r\n val o = js("{}")\r\n\r\n
o["type"] = type\r\n o["request"] = request\r\n o["response"] = response\r\n o["options"] = options\r\n\r\n
return o\r\n}\r\n\r\n/**\r\n * Exposes the JavaScript
[CacheStorage](https://developer.mozilla.org/en/docs/Web/API/CacheStorage) to Kotlin\r\n */\r\npublic external
abstract class CacheStorage {\r\n fun match(request: dynamic, options: CacheQueryOptions = definedExternally):
Promise<Any?>\r\n fun has(cacheName: String): Promise<Boolean>\r\n fun open(cacheName: String):
Promise<Cache>\r\n fun delete(cacheName: String): Promise<Boolean>\r\n fun keys():
Promise<dynamic>\r\n}\r\n\r\npublic external open class FunctionalEvent : ExtendableEvent {\r\n}\r\n\r\npublic
external @marker interface UnionClientOrMessagePortOrServiceWorker {\r\n}\r\n\r\n/* please, don't implement
this interface! */\r\npublic external interface ServiceWorkerState {\r\n companion object\r\n}\r\n\r\npublic inline val
ServiceWorkerState.Companion.INSTALLING: ServiceWorkerState get() =
"installing".asDynamic().unsafeCast<ServiceWorkerState>()\r\n\r\npublic inline val
ServiceWorkerState.Companion.INSTALLED: ServiceWorkerState get() =
"installed".asDynamic().unsafeCast<ServiceWorkerState>()\r\n\r\npublic inline val
ServiceWorkerState.Companion.ACTIVATING: ServiceWorkerState get() =
"activating".asDynamic().unsafeCast<ServiceWorkerState>()\r\n\r\npublic inline val
ServiceWorkerState.Companion.ACTIVATED: ServiceWorkerState get() =
"activated".asDynamic().unsafeCast<ServiceWorkerState>()\r\n\r\npublic inline val
ServiceWorkerState.Companion.REDUNDANT: ServiceWorkerState get() =
"redundant".asDynamic().unsafeCast<ServiceWorkerState>()\r\n\r\n/* please, don't implement this interface!
*/\r\n\r\npublic external interface FrameType {\r\n companion object\r\n}\r\n\r\npublic inline val
FrameType.Companion.AUXILIARY: FrameType get() =
"auxiliary".asDynamic().unsafeCast<FrameType>()\r\n\r\npublic inline val FrameType.Companion.TOP_LEVEL:
FrameType get() = "top-level".asDynamic().unsafeCast<FrameType>()\r\n\r\npublic inline val
FrameType.Companion.NESTED: FrameType get() = "nested".asDynamic().unsafeCast<FrameType>()\r\n\r\npublic
inline val FrameType.Companion.NONE: FrameType get() =
"none".asDynamic().unsafeCast<FrameType>()\r\n\r\n/* please, don't implement this interface! */\r\n\r\npublic
external interface ClientType {\r\n companion object\r\n}\r\n\r\npublic inline val ClientType.Companion.WINDOW:
ClientType get() = "window".asDynamic().unsafeCast<ClientType>()\r\n\r\npublic inline val
ClientType.Companion.WORKER: ClientType get() = "worker".asDynamic().unsafeCast<ClientType>()\r\n\r\npublic
inline val ClientType.Companion.SHAREDWORKER: ClientType get() =
"sharedworker".asDynamic().unsafeCast<ClientType>()\r\n\r\npublic inline val ClientType.Companion.ALL:
ClientType get() = "all".asDynamic().unsafeCast<ClientType>()\r\n\r\n", "/*\r\n * Copyright 2010-2018 JetBrains
s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0
license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n// NOTE: THIS FILE IS AUTO-
GENERATED, DO NOT EDIT!\r\n\r\n// See libraries/tools/idl2k for
details\r\n\r\n@file:Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\r\n\r\npackage
org.w3c.xhr\r\n\r\nimport kotlin.js.*\r\nimport org.khronos.webgl.*\r\nimport org.w3c.css.masking.*\r\nimport
org.w3c.dom.*\r\nimport org.w3c.dom.clipboard.*\r\nimport org.w3c.dom.css.*\r\nimport
org.w3c.dom.events.*\r\nimport org.w3c.dom.mediacapture.*\r\nimport org.w3c.dom.parsing.*\r\nimport
org.w3c.dom.pointerevents.*\r\nimport org.w3c.dom.svg.*\r\nimport org.w3c.dom.url.*\r\nimport

```





```

TYPEALIAS\r\n)\r\n@MustBeDocumented\r\n@ExperimentalStdlibApi\r\n@SinceKotlin("1.3") // TODO:
Remove experimental status from itself and advance SinceKotlin to 1.4\r\npublic annotation class
ExperimentalStdlibApi\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin\r\n\r\nimport kotlin.annotation.AnnotationTarget.*\r\nimport
kotlin.experimental.ExperimentalTypeInference\r\n\r\n/**\r\n * Allows to infer generic type arguments of a function
from the calls in the annotated function parameter of that function.\r\n */\r\n * When this annotation is placed on a
generic function parameter of a function,\r\n * it enables to infer the type arguments of that generic function from
the lambda body passed to that parameter.\r\n */\r\n * The calls that affect inference are either members of the
receiver type of an annotated function parameter or\r\n * extensions for that type. The extensions must be
themselves annotated with `@BuilderInference`.\r\n */\r\n * Example: we declare\r\n * ```\r\n * fun <T>
sequence(@BuilderInference block: suspend SequenceScope<T>().-> Unit): Sequence<T>\r\n * ```\r\n * and use it
like\r\n * ```\r\n * val result = sequence { yield("result") }\r\n * ```\r\n * Here the type argument of the resulting
sequence is inferred to `String` from\r\n * the argument of the [SequenceScope.yield] function, that is called inside
the lambda passed to [sequence].\r\n */\r\n * Note: this annotation is experimental, see [ExperimentalTypeInference]
on how to opt-in for it.\r\n */\r\n @Target(VALUE_PARAMETER, FUNCTION,
PROPERTY)\r\n @Retention(AnnotationRetention.BINARY)\r\n @SinceKotlin("1.3")\r\n @ExperimentalTypeInfe
rence\r\n public annotation class BuilderInference\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin\r\n\r\nimport
kotlin.annotation.AnnotationTarget.*\r\nimport kotlin.internal.RequireKotlin\r\nimport
kotlin.internal.RequireKotlinVersionKind\r\n\r\n/**\r\n * The experimental multiplatform support API marker.\r\n
*\r\n * Any usage of a declaration annotated with `@ExperimentalMultiplatform` must be accepted either by\r\n *
annotating that usage with the [UseExperimental] annotation, e.g.
`@UseExperimental(ExperimentalMultiplatform::class)`,\r\n * or by using the compiler argument `-Xuse-
experimental=kotlin.ExperimentalMultiplatform`.\r\n
*\r\n @Experimental\r\n @MustBeDocumented\r\n @Target(\r\n CLASS,\r\n ANNOTATION_CLASS,\r\n
PROPERTY,\r\n FIELD,\r\n LOCAL_VARIABLE,\r\n VALUE_PARAMETER,\r\n CONSTRUCTOR,\r\n
FUNCTION,\r\n PROPERTY_GETTER,\r\n PROPERTY_SETTER,\r\n
TYPEALIAS)\r\n)\r\n @Retention(AnnotationRetention.BINARY)\r\n @RequireKotlin("1.2.50", versionKind =
RequireKotlinVersionKind.COMPILER_VERSION)\r\n public annotation class
ExperimentalMultiplatform\r\n\r\n/**\r\n * Marks an expected annotation class that it isn't required to have actual
counterparts in all platforms.\r\n */\r\n * This annotation is only applicable to `expect` annotation classes in multi-
platform projects and marks that class as `optional`.\r\n * Optional expected class is allowed to have no
corresponding actual class on the platform. Optional annotations can only be used\r\n * to annotate something, not
as types in signatures. If an optional annotation has no corresponding actual class on a platform,\r\n * the annotation
entries where it's used are simply erased when compiling code on that platform.\r\n */\r\n * Note: this annotation is
experimental, see [ExperimentalMultiplatform] on how to opt-in for it.\r\n
*\r\n @Target(ANNOTATION_CLASS)\r\n @Retention(AnnotationRetention.BINARY)\r\n @ExperimentalMultipl
atform\r\n @RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)\r\n public
annotation class OptionalExpectation\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.collections\r\n\r\nimport kotlin.js.JsName\r\n\r\n/**\r\n *
Provides a skeletal implementation of the read-only [Collection] interface.\r\n */\r\n * @param E the type of
elements contained in the collection. The collection is covariant on its element type.\r\n
*\r\n @SinceKotlin("1.1")\r\n public abstract class AbstractCollection<out E> protected constructor() :
Collection<E> {\r\n abstract override val size: Int\r\n abstract override fun iterator(): Iterator<E>\r\n\r\n
override fun contains(element: @UnsafeVariance E): Boolean = any { it == element }\r\n\r\n override fun

```

```

containsAll(elements: Collection<@UnsafeVariance E>): Boolean =\r\n elements.all { contains(it) } // use
when js will support bound refs: elements.all(this::contains)\r\n\r\n override fun isEmpty(): Boolean = size ==
0\r\n\r\n override fun toString(): String = joinToString(", ", "[", "]") {\r\n if (it === this) \"(this
Collection)\" else it.toString()\r\n }\r\n\r\n /**\r\n * Returns new array of type `Array<Any?>` with the
elements of this collection.\r\n */\r\n @JsName(\"toArray\")\r\n protected open fun toArray(): Array<Any?> =
copyToArrayImpl(this)\r\n\r\n /**\r\n * Fills the provided [array] or creates new array of the same type\r\n * and fills it with the elements of this collection.\r\n */\r\n protected open fun <T> toArray(array: Array<T>):
Array<T> = copyToArrayImpl(this, array)\r\n\r\n\"\", \"\"/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.collections\r\n\r\nprivate enum class State
{\r\n Ready,\r\n NotReady,\r\n Done,\r\n Failed\r\n}\r\n\r\n/**\r\n * A base class to simplify implementing
iterators so that implementations only have to implement [computeNext]\r\n * to implement the iterator, calling
[done] when the iteration is complete.\r\n */\r\npublic abstract class AbstractIterator<T> : Iterator<T> {\r\n private
var state = State.NotReady\r\n private var nextValue: T? = null\r\n\r\n override fun hasNext(): Boolean {\r\n require(state != State.Failed)\r\n return when (state) {\r\n State.Done -> false\r\n State.Ready ->
true\r\n else -> tryComputeNext()\r\n }\r\n\r\n override fun next(): T {\r\n if (!hasNext())
throw NoSuchElementException\r\n state = State.NotReady\r\n @Suppress(\"UNCHECKED_CAST\")\r\n return nextValue as T\r\n }\r\n\r\n private fun
tryComputeNext(): Boolean {\r\n state = State.Failed\r\n computeNext()\r\n return state ==
State.Ready\r\n }\r\n\r\n /**\r\n * Computes the next item in the iterator.\r\n */\r\n * This callback method
should call one of these two methods:\r\n * * [setNext] with the next value of the iteration\r\n * *
[done] to indicate there are no more elements\r\n * * Failure to call either method will result in the iteration
terminating with a failed state\r\n */\r\n abstract protected fun computeNext(): Unit\r\n\r\n /**\r\n * Sets the
next value in the iteration, called from the [computeNext] function\r\n */\r\n protected fun setNext(value: T):
Unit {\r\n nextValue = value\r\n state = State.Ready\r\n }\r\n\r\n /**\r\n * Sets the state to done so
that the iteration terminates.\r\n */\r\n protected fun done() {\r\n state = State.Done\r\n }\r\n\r\n}\r\n\r\n\"\", \"\"/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt
file.\r\n */\r\n\r\n/*\r\n * Based on GWT AbstractList\r\n * Copyright 2007 Google Inc.\r\n */\r\n\r\npackage
kotlin.collections\r\n\r\n/**\r\n * Provides a skeletal implementation of the read-only [List] interface.\r\n */\r\n * This
class is intended to help implementing read-only lists so it doesn't support concurrent modification tracking.\r\n
*\r\n * @param E the type of elements contained in the list. The list is covariant on its element type.\r\n
*\r\n */\r\n@SinceKotlin(\"1.1\")\r\npublic abstract class AbstractList<out E> protected constructor() :
AbstractCollection<E>(), List<E> {\r\n abstract override val size: Int\r\n abstract override fun get(index: Int):
E\r\n\r\n override fun iterator(): Iterator<E> = IteratorImpl()\r\n\r\n override fun indexOf(element:
@UnsafeVariance E): Int = indexOfFirst { it == element }\r\n\r\n override fun lastIndexOf(element:
@UnsafeVariance E): Int = indexOfLast { it == element }\r\n\r\n override fun listIterator(): ListIterator<E> =
ListIteratorImpl(0)\r\n\r\n override fun listIterator(index: Int): ListIterator<E> = ListIteratorImpl(index)\r\n\r\n
override fun subList(fromIndex: Int, toIndex: Int): List<E> = SubList(this, fromIndex, toIndex)\r\n\r\n private
class SubList<out E>(private val list: AbstractList<E>, private val fromIndex: Int, toIndex: Int) : AbstractList<E>(),
RandomAccess {\r\n private var _size: Int = 0\r\n\r\n init {\r\n checkRangeIndexes(fromIndex,
toIndex, list.size)\r\n this._size = toIndex - fromIndex\r\n }\r\n\r\n override fun get(index: Int): E
{\r\n checkElementIndex(index, _size)\r\n return list[fromIndex + index]\r\n }\r\n\r\n
override val size: Int get() = _size\r\n }\r\n\r\n /**\r\n * Compares this list with other list instance with the
ordered structural equality.\r\n */\r\n * @return true, if [other] instance is a [List] of the same size, which
contains the same elements in the same order.\r\n */\r\n override fun equals(other: Any?): Boolean {\r\n if
(other === this) return true\r\n if (other !is List<*>) return false\r\n\r\n return orderedEquals(this, other)\r\n }\r\n\r\n /**\r\n * Returns the hash code value for this list.\r\n */\r\n override fun hashCode(): Int =

```

```

orderedHashCode(this)\r\n\r\n private open inner class IteratorImpl : Iterator<E> {\r\n /** the index of the
item that will be returned on the next call to [next]`() ` */\r\n protected var index = 0\r\n\r\n override fun
hasNext(): Boolean = index < size\r\n\r\n override fun next(): E {\r\n if (!hasNext()) throw
NoSuchElementException()\r\n return get(index++)\r\n }\r\n }\r\n\r\n /**\r\n * Implementation of
[ListIterator] for abstract lists.\r\n */\r\n private open inner class ListIteratorImpl(index: Int) : IteratorImpl(),
ListIterator<E> {\r\n\r\n init {\r\n checkPositionIndex(index, this@AbstractList.size)\r\n
this.index = index\r\n }\r\n\r\n override fun hasPrevious(): Boolean = index > 0\r\n\r\n override fun
nextIndex(): Int = index\r\n\r\n override fun previous(): E {\r\n if (!hasPrevious()) throw
NoSuchElementException()\r\n return get(--index)\r\n }\r\n\r\n override fun previousIndex(): Int =
index - 1\r\n }\r\n\r\n internal companion object {\r\n internal fun checkElementIndex(index: Int, size: Int)
{\r\n if (index < 0 || index >= size) {\r\n throw IndexOutOfBoundsException("\r\nindex: $index, size:
$size")\r\n }\r\n }\r\n\r\n internal fun checkPositionIndex(index: Int, size: Int) {\r\n if (index
< 0 || index > size) {\r\n throw IndexOutOfBoundsException("\r\nindex: $index, size: $size")\r\n }\r\n
 }\r\n\r\n internal fun checkRangeIndexes(fromIndex: Int, toIndex: Int, size: Int) {\r\n if (fromIndex <
0 || toIndex > size) {\r\n throw IndexOutOfBoundsException("\r\nfromIndex: $fromIndex, toIndex: $toIndex,
size: $size")\r\n }\r\n if (fromIndex > toIndex) {\r\n throw
IllegalArgumentOutOfRangeException("\r\nfromIndex: $fromIndex > toIndex: $toIndex")\r\n }\r\n }\r\n\r\n internal fun checkBoundsIndexes(startIndex: Int, endIndex: Int, size: Int) {\r\n if (startIndex < 0 || endIndex >
size) {\r\n throw IndexOutOfBoundsException("\r\nstartIndex: $startIndex, endIndex: $endIndex, size:
$size")\r\n }\r\n if (startIndex > endIndex) {\r\n throw
IllegalArgumentOutOfRangeException("\r\nstartIndex: $startIndex > endIndex: $endIndex")\r\n }\r\n }\r\n\r\n internal fun orderedHashCode(c: Collection<*>): Int {\r\n var hashCode = 1\r\n for (e in c) {\r\n
 hashCode = 31 * hashCode + (e?.hashCode() ?: 0)\r\n }\r\n return hashCode\r\n }\r\n\r\n internal fun orderedEquals(c: Collection<*>, other: Collection<*>): Boolean {\r\n if (c.size != other.size)
return false\r\n\r\n val otherIterator = other.iterator()\r\n for (elem in c) {\r\n val elemOther =
otherIterator.next()\r\n if (elem != elemOther) {\r\n return false\r\n }\r\n }\r\n return true\r\n }\r\n }\r\n\r\n /**\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n /**\r\n * Based on GWT AbstractMap\r\n * Copyright 2007 Google Inc.\r\n */\r\n /**\r\n * Package kotlin.collections\r\n */\r\n /**\r\n * Provides a skeletal implementation of the read-only [Map]
interface.\r\n */\r\n /**\r\n * The implementor is required to implement [entries] property, which should return read-only set
of map entries.\r\n */\r\n /**\r\n * @param K the type of map keys. The map is invariant on its key type.\r\n */\r\n /**\r\n * @param V the type of map values. The map is covariant on its value type.\r\n */\r\n @SinceKotlin("1.1")\r\n public abstract
class AbstractMap<K, out V> protected constructor() : Map<K, V> {\r\n\r\n override fun containsKey(key: K):
Boolean {\r\n return implFindEntry(key) != null\r\n }\r\n\r\n override fun containsValue(value:
@UnsafeVariance V): Boolean = entries.any { it.value == value }\r\n\r\n internal fun containsEntry(entry:
Map.Entry<*, *>): Boolean {\r\n // since entry comes from @UnsafeVariance parameters it can be virtually
anything\r\n if (entry !is Map.Entry<*, *>) return false\r\n val key = entry.key\r\n val value =
entry.value\r\n val ourValue = get(key)\r\n if (value != ourValue) {\r\n return false\r\n
 }\r\n\r\n // Perhaps it was null and we don't contain the key?\r\n if (ourValue == null &&
!containsKey(key)) {\r\n return false\r\n }\r\n return true\r\n }\r\n\r\n /**\r\n *
Compares this map with other instance with the ordered structural equality.\r\n */\r\n /**\r\n * @return true, if [other]
instance is a [Map] of the same size, all entries of which are contained in the [entries] set of this map.\r\n */\r\n override fun equals(other: Any?): Boolean {\r\n if (other === this) return true\r\n if (other !is Map<*, *>)
return false\r\n if (size != other.size) return false\r\n\r\n return other.entries.all { containsEntry(it) }\r\n
 }\r\n\r\n override operator fun get(key: K): V? = implFindEntry(key)?.value\r\n\r\n /**\r\n * Returns the
hash code value for this map.\r\n */\r\n /**\r\n * It is the same as the hashCode of [entries] set.\r\n */\r\n override
fun hashCode(): Int = entries.hashCode()\r\n\r\n override fun isEmpty(): Boolean = size == 0\r\n override val

```

```

size: Int get() = entries.size\r\n\r\n /**\r\n * Returns a read-only [Set] of all keys in this map.\r\n *\r\n * Accessing this property first time creates a keys view from [entries].\r\n * All subsequent accesses just return the
created instance.\r\n */\r\n override val keys: Set<K>\r\n get() {\r\n if (_keys == null) {\r\n _keys = object : AbstractSet<K>() {\r\n override operator fun contains(element: K): Boolean =
containsKey(element)\r\n\r\n override operator fun iterator(): Iterator<K> {\r\n val
entryIterator = entries.iterator()\r\n return object : Iterator<K> {\r\n override fun
hasNext(): Boolean = entryIterator.hasNext()\r\n override fun next(): K =
entryIterator.next().key\r\n }\r\n }\r\n\r\n override val size: Int get() =
this@AbstractMap.size\r\n }\r\n }\r\n return _keys!!\r\n }\r\n\r\n @kotlin.jvm.Volatile\r\n private var _keys: Set<K>? = null\r\n\r\n override fun toString(): String =
entries.joinToString(", ", "\n", "\n") { toString(it) }\r\n\r\n private fun toString(entry: Map.Entry<K, V>): String =
toString(entry.key) + "\n" + toString(entry.value)\r\n\r\n private fun toString(o: Any?): String = if (o === this)
"\n(this Map)\n" else o.toString()\r\n\r\n /**\r\n * Returns a read-only [Collection] of all values in this map.\r\n *\r\n * Accessing this property first time creates a values view from [entries].\r\n * All subsequent accesses just
return the created instance.\r\n */\r\n override val values: Collection<V>\r\n get() {\r\n if (_values ==
null) {\r\n _values = object : AbstractCollection<V>() {\r\n override operator fun
contains(element: @UnsafeVariance V): Boolean = containsValue(element)\r\n\r\n override operator fun
iterator(): Iterator<V> {\r\n val entryIterator = entries.iterator()\r\n return object :
Iterator<V> {\r\n override fun hasNext(): Boolean = entryIterator.hasNext()\r\n override
fun next(): V = entryIterator.next().value\r\n }\r\n }\r\n\r\n override
val size: Int get() = this@AbstractMap.size\r\n }\r\n }\r\n return _values!!\r\n }\r\n\r\n @kotlin.jvm.Volatile\r\n private var _values: Collection<V>? = null\r\n\r\n private fun implFindEntry(key: K):
Map.Entry<K, V>? = entries.firstOrNull { it.key == key }\r\n\r\n internal companion object {\r\n internal
fun entryHashCode(e: Map.Entry<*, *>): Int = with(e) { (key?.hashCode() ?: 0) xor (value?.hashCode() ?: 0) }\r\n\r\n internal fun
entryToString(e: Map.Entry<*, *>): String = with(e) { "\nkey=$value" }\r\n\r\n internal fun
entryEquals(e: Map.Entry<*, *>, other: Any?): Boolean {\r\n if (other !is Map.Entry<*, *>) return false\r\n
 return e.key == other.key && e.value == other.value\r\n }\r\n }\r\n\r\n /**\r\n * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage
kotlin.collections\r\n\r\n/**\r\n * Provides a skeletal implementation of the read-only [Set] interface.\r\n *\r\n * This
class is intended to help implementing read-only sets so it doesn't support concurrent modification tracking.\r\n *\r\n * @param E the type of elements contained in the set. The set is covariant on its element type.\r\n */\r\n\r\n@SinceKotlin("1.1")\r\npublic abstract class AbstractSet<out E> protected constructor() :
AbstractCollection<E>(), Set<E> {\r\n\r\n /**\r\n * Compares this set with other set instance with the unordered
structural equality.\r\n *\r\n * @return true, if [other] instance is a [Set] of the same size, all elements of which
are contained in this set.\r\n */\r\n override fun equals(other: Any?): Boolean {\r\n if (other === this) return
true\r\n if (other !is Set<*>) return false\r\n return setEquals(this, other)\r\n }\r\n\r\n /**\r\n * Returns
the hash code value for this set.\r\n */\r\n override fun hashCode(): Int = unorderedHashCode(this)\r\n\r\n internal companion object {\r\n internal fun unorderedHashCode(c: Collection<*>): Int {\r\n var
hashCode = 0\r\n for (element in c) {\r\n hashCode += (element?.hashCode() ?: 0)\r\n }\r\n
 return hashCode\r\n }\r\n\r\n internal fun setEquals(c: Set<*>, other: Set<*>): Boolean {\r\n if
(c.size != other.size) return false\r\n return c.containsAll(other)\r\n }\r\n }\r\n\r\n /**\r\n * Copyright
2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is
governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("ArraysKt")\r\n\r\npackage
kotlin.collections\r\n\r\nimport kotlin.contracts.*\r\n\r\n/**\r\n * Returns a single list of all elements from all
arrays in the given array.\r\n * @sample samples.collections.Arrays.Transformations.flattenArray\r\n */\r\n\r\npublic
fun <T> Array<out Array<out T>>.flatten(): List<T> {\r\n val result = ArrayList<T>(sumBy { it.size })\r\n for

```

```

(element in this) {
 result.addAll(element)
}
return result
}
Returns a pair of lists, where *first* list is built from the first values of each pair from this array, *second* list is built from the second values of each pair from this array.
@sample
samples.collections.Arrays.Transformations.unzipArray
public fun <T, R> Array<out Pair<T, R>>.unzip(): Pair<List<T>, List<R>> {
 val listT = ArrayList<T>(size)
 val listR = ArrayList<R>(size)
 for (pair in this) {
 listT.add(pair.first)
 listR.add(pair.second)
 }
 return listT to listR
}
Returns `true` if this nullable array is either null or empty.
@sample
samples.collections.Arrays.Usage.arrayIsNullOrEmpty
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun Array<*>?.isNullOrEmpty(): Boolean {
 contract {
 returns(false) implies (this@isNullOrEmpty != null)
 }
 return this == null || this.isEmpty()
}
Returns this array if it's not empty or the result of calling [defaultValue] function if the array is empty.
@sample
samples.collections.Arrays.Usage.arrayIfEmpty
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("UPPER_BOUND_CANNOT_BE_ARRAY")
public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : Array<*>, C : R = {
 if (isEmpty()) defaultValue() else
this
}
@UseExperimental(ExperimentalUnsignedTypes::class)
@SinceKotlin("1.3")
@PublishedApi
@kotlin.jvm.JvmName("contentDeepEquals")
@kotlin.js.JsName("contentDeepEqualsImpl")
internal fun <T> Array<out T>.contentDeepEqualsImpl(other: Array<out T>): Boolean {
 if (this === other) return true
 if (this.size != other.size) return false
 for (i in indices) {
 val v1 = this[i]
 val v2 = other[i]
 if (v1 === v2) continue
 else if (v1 == null || v2 == null) return false
 }
 when {
 v1 is Array<*> && v2 is Array<*> -> if (!v1.contentDeepEquals(v2)) return false
 v1 is ByteArray && v2 is ByteArray -> if (!v1.contentEquals(v2)) return false
 v1 is ShortArray && v2 is ShortArray -> if (!v1.contentEquals(v2)) return false
 v1 is IntArray && v2 is IntArray -> if (!v1.contentEquals(v2)) return false
 v1 is LongArray && v2 is LongArray -> if (!v1.contentEquals(v2)) return false
 v1 is FloatArray && v2 is FloatArray -> if (!v1.contentEquals(v2)) return false
 v1 is DoubleArray && v2 is DoubleArray -> if (!v1.contentEquals(v2)) return false
 v1 is CharArray && v2 is CharArray -> if (!v1.contentEquals(v2)) return false
 v1 is BooleanArray && v2 is BooleanArray -> if (!v1.contentEquals(v2)) return false
 v1 is UByteArray && v2 is UByteArray -> if (!v1.contentEquals(v2)) return false
 v1 is UShortArray && v2 is UShortArray -> if (!v1.contentEquals(v2)) return false
 v1 is UIntArray && v2 is UIntArray -> if (!v1.contentEquals(v2)) return false
 v1 is ULongArray && v2 is ULongArray -> if (!v1.contentEquals(v2)) return false
 }
 else -> if (v1 != v2) return false
}
return true
}
@SinceKotlin("1.3")
@PublishedApi
@kotlin.jvm.JvmName("contentDeepToString")
@kotlin.js.JsName("contentDeepToStringImpl")
internal fun <T> Array<out T>.contentDeepToStringImpl(): String {
 val length = size.coerceAtMost((Int.MAX_VALUE - 2) / 5) * 5 + 2 // in order not to overflow Int.MAX_VALUE
 return buildString(length) {
 contentDeepToStringInternal(this, mutableListOf())
 }
}
@UseExperimental(ExperimentalUnsignedTypes::class)
private fun <T> Array<out T>.contentDeepToStringInternal(result: StringBuilder, processed: MutableList<Array<*>>) {
 if (this in processed) {
 result.append("[...]")
 return
 }
 processed.add(this)
 result.append("[")
 for (i in indices) {
 if (i != 0) {
 result.append(", ")
 }
 val element = this[i]
 when (element) {
 null -> result.append("null")
 is Array<*> -> element.contentDeepToStringInternal(result, processed)
 is ByteArray -> result.append(element.contentToString())
 is ShortArray -> result.append(element.contentToString())
 is IntArray -> result.append(element.contentToString())
 is LongArray -> result.append(element.contentToString())
 is FloatArray -> result.append(element.contentToString())
 is DoubleArray -> result.append(element.contentToString())
 is CharArray ->

```



```

result.append(element.toString())\r\n is BooleanArray ->
result.append(element.toString())\r\n\r\n is UByteArray ->
result.append(element.toString())\r\n is UShortArray -> result.append(element.toString())\r\n
 is UIntArray -> result.append(element.toString())\r\n is ULongArray ->
result.append(element.toString())\r\n\r\n else -> result.append(element.toString())\r\n }\r\n
}\r\n\r\n result.append('}')\r\n processed.removeAt(processed.lastIndexOf('}')\r\n}", "/*\r\n * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n\r\npackage
kotlin.collections\r\n\r\n/**\r\n * Data class representing a value from a collection or sequence, along with its index
in that collection or sequence.\r\n * \r\n * @property value the underlying value.\r\n * @property index the index of
the value in the collection or sequence.\r\n */\r\n\r\npublic data class IndexedValue<out T>(public val index: Int, public
val value: T)\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt
file.\r\n */\r\n\r\n\r\n@file:kotlin.jvm.JvmName("MapAccessorsKt")\r\n\r\n\r\npackage kotlin.collections\r\n\r\n\r\nimport
kotlin.reflect.KProperty\r\n\r\nimport kotlin.internal.Exact\r\n\r\n\r\n/**\r\n * Returns the value of the property for the
given object from this read-only map.\r\n * @param thisRef the object for which the value is requested (not
used).\r\n * @param property the metadata for the property, used to get the name of property and lookup the value
corresponding to this name in the map.\r\n * @return the property value.\r\n * \r\n * @throws
NoSuchElementException when the map doesn't contain value for the property name and doesn't provide an implicit
default (see [withDefault]).\r\n */\r\n\r\n@kotl.in.internal.InlineOnly\r\n\r\npublic inline operator fun <V, V1 : V> Map<in
String, @Exact V>.getValue(thisRef: Any?, property: KProperty<*>): V1 =\r\n
@Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)\r\n\r\n\r\n\r\n/**\r\n * Returns the
value of the property for the given object from this mutable map.\r\n * @param thisRef the object for which the
value is requested (not used).\r\n * @param property the metadata for the property, used to get the name of property
and lookup the value corresponding to this name in the map.\r\n * @return the property value.\r\n * \r\n * @throws
NoSuchElementException when the map doesn't contain value for the property name and doesn't provide an implicit
default (see [withDefault]).\r\n */\r\n\r\n@kotl.in.jvm.JvmName("getVar")\r\n\r\n@kotl.in.internal.InlineOnly\r\n\r\npublic
inline operator fun <V, V1 : V> MutableMap<in String, out @Exact V>.getValue(thisRef: Any?, property:
KProperty<*>): V1 =\r\n
@Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as
V1)\r\n\r\n\r\n@Deprecated("Use getValue() with two type parameters instead", level =
DeprecationLevel.ERROR)\r\n\r\n@kotl.in.jvm.JvmName("getVarContravariant")\r\n\r\n@kotl.in.internal.LowPriorityInO
verloadResolution\r\n\r\n@kotl.in.internal.InlineOnly\r\n\r\npublic inline fun <V> MutableMap<in String, in
V>.getValue(thisRef: Any?, property: KProperty<*>): V =\r\n
@Suppress("UNCHECKED_CAST")
(getOrImplicitDefault(property.name) as V)\r\n\r\n\r\n\r\n/**\r\n * Stores the value of the property for the given object in
this mutable map.\r\n * @param thisRef the object for which the value is requested (not used).\r\n * @param
property the metadata for the property, used to get the name of property and store the value associated with that
name in the map.\r\n * @param value the value to set.\r\n */\r\n\r\n@kotl.in.internal.InlineOnly\r\n\r\npublic inline operator
fun <V> MutableMap<in String, in V>.setValue(thisRef: Any?, property: KProperty<*>, value: V) {\r\n
this.put(property.name, value)\r\n}\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n
*/\r\n\r\n\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n\r\n@file:kotlin.jvm.JvmName("MapsKt")\r\n\r\n\r\n\r\npackage
kotlin.collections\r\n\r\n\r\n\r\n\r\n\r\n/**\r\n * Returns the value for the given key, or the implicit default value for this map.\r\n *
By default no implicit value is provided for maps and a [NoSuchElementException] is thrown.\r\n * To create a map
with implicit default value use [withDefault] method.\r\n * \r\n * @throws NoSuchElementException when the map
doesn't contain a value for the specified key and no implicit default was provided for that map.\r\n */\r\n
\r\n\r\n@kotl.in.jvm.JvmName("getOrImplicitDefaultNullable")\r\n\r\n@PublishedApi\r\n\r\n\r\ninternal fun <K, V> Map<K,
V>.getOrImplicitDefault(key: K): V {\r\n
if (this is MapWithDefault)\r\n
return

```

```

this.getOrImplicitDefault(key)\r\n\r\n return getOrElseNullable(key, { throw NoSuchElementException("Key
$key is missing in the map.") })\r\n}\r\n\r\n/**\r\n * Returns a wrapper of this read-only map, having the implicit
default value provided with the specified function [defaultValue].\r\n * This implicit default value is used when
the original map doesn't contain a value for the key specified\r\n * and a value is obtained with [Map.getValue]
function, for example when properties are delegated to the map.\r\n * When this map already has an implicit
default value provided with a former call to [withDefault], it is being replaced by this call.\r\n */\r\npublic fun <K,
V> Map<K, V>.withDefault(defaultValue: (key: K) -> V): Map<K, V> =\r\n when (this) {\r\n is
MapWithDefault -> this.map.withDefault(defaultValue)\r\n else -> MapWithDefaultImpl(this, defaultValue)\r\n }
\r\n\r\n/**\r\n * Returns a wrapper of this mutable map, having the implicit default value provided with the
specified function [defaultValue].\r\n * This implicit default value is used when the original map doesn't
contain a value for the key specified\r\n * and a value is obtained with [Map.getValue] function, for example when
properties are delegated to the map.\r\n * When this map already has an implicit default value provided with a
former call to [withDefault], it is being replaced by this call.\r\n */\r\npublic fun <K, V> MutableMap<K,
V>.withDefault(defaultValue: (key: K) -> V): MutableMap<K, V> =\r\n when (this) {\r\n is
MutableMapWithDefault -> this.map.withDefault(defaultValue)\r\n else -> MutableMapWithDefaultImpl(this,
defaultValue)\r\n }\r\n\r\nprivate interface MapWithDefault<K, out V> : Map<K, V> {\r\n public val map:
Map<K, V>\r\n public fun getOrImplicitDefault(key: K): V\r\n}\r\n\r\nprivate interface
MutableMapWithDefault<K, V> : MutableMap<K, V>, MapWithDefault<K, V> {\r\n public override val map:
MutableMap<K, V>\r\n}\r\n\r\nprivate class MapWithDefaultImpl<K, out V>(public override val map: Map<K,
V>, private val default: (key: K) -> V) : MapWithDefault<K, V> {\r\n override fun equals(other: Any?): Boolean
= map.equals(other)\r\n override fun hashCode(): Int = map.hashCode()\r\n override fun toString(): String =
map.toString()\r\n override val size: Int get() = map.size\r\n override fun isEmpty(): Boolean =
map.isEmpty()\r\n override fun containsKey(key: K): Boolean = map.containsKey(key)\r\n override fun
containsValue(value: @UnsafeVariance V): Boolean = map.containsValue(value)\r\n override fun get(key: K): V?
= map.get(key)\r\n override val keys: Set<K> get() = map.keys\r\n override val values: Collection<V> get() =
map.values\r\n override val entries: Set<Map.Entry<K, V>> get() = map.entries\r\n\r\n override fun
getOrImplicitDefault(key: K): V = map.getOrElseNullable(key, { default(key) })\r\n}\r\n\r\nprivate class
MutableMapWithDefaultImpl<K, V>(public override val map: MutableMap<K, V>, private val default: (key: K) ->
V) : MutableMapWithDefault<K, V> {\r\n override fun equals(other: Any?): Boolean = map.equals(other)\r\n
 override fun hashCode(): Int = map.hashCode()\r\n override fun toString(): String = map.toString()\r\n
 override val size: Int get() = map.size\r\n override fun isEmpty(): Boolean = map.isEmpty()\r\n override fun
containsKey(key: K): Boolean = map.containsKey(key)\r\n override fun containsValue(value: @UnsafeVariance
V): Boolean = map.containsValue(value)\r\n override fun get(key: K): V? = map.get(key)\r\n override val keys:
MutableSet<K> get() = map.keys\r\n override val values: MutableCollection<V> get() = map.values\r\n override
val entries: MutableSet<MutableMap.MutableEntry<K, V>> get() = map.entries\r\n\r\n override fun put(key: K,
value: V): V? = map.put(key, value)\r\n override fun remove(key: K): V? = map.remove(key)\r\n override fun
putAll(from: Map<out K, V>) = map.putAll(from)\r\n override fun clear() = map.clear()\r\n\r\n override fun
getOrImplicitDefault(key: K): V = map.getOrElseNullable(key, { default(key) })\r\n}\r\n\r\n"/**\r\n * Copyright
2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is
governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("CollectionsKt")\r\n\r\npackage
kotlin.collections\r\n\r\nimport kotlin.random.Random\r\n\r\n/**\r\n * Removes a single instance of the specified
element from this\r\n * collection, if it is present.\r\n * Allows to overcome type-safety restriction of `remove`
that requires to pass an element of type `E`.\r\n * @return `true` if the element has been successfully removed;
`false` if it was not present in the collection.\r\n */\r\n@kotlin.internal.InlineOnly\r\npublic inline fun
<@kotlin.internal.InputTypes T> MutableCollection<out T>.remove(element: T): Boolean =\r\n @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).remove(element)\r\n\r\n/**\r\n * Removes

```

all of this collection's elements that are also contained in the specified collection.

\* Allows to overcome type-safety restriction of `removeAll` that requires to pass a collection of type `Collection<E>`. @return `true` if any of the specified elements was removed from the collection, `false` if the collection was not modified.

```

@kotlin.internal.InlineOnly
public inline fun <@kotlin.internal.OnlyInputTypes T> MutableCollection<out T>.removeAll(elements: Collection<T>): Boolean = @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).removeAll(elements)

```

\* Retains only the elements in this collection that are contained in the specified collection.

\* Allows to overcome type-safety restriction of `retainAll` that requires to pass a collection of type `Collection<E>`. @return `true` if any element was removed from the collection, `false` if the collection was not modified.

```

@kotlin.internal.InlineOnly
public inline fun <@kotlin.internal.OnlyInputTypes T> MutableCollection<out T>.retainAll(elements: Collection<T>): Boolean = @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).retainAll(elements)

```

Removes the element at the specified [index] from this list.

\* In Kotlin one should use the [MutableList.removeAt] function instead.

```

@Deprecated("Use removeAt(index) instead.", ReplaceWith("removeAt(index)"), level = DeprecationLevel.ERROR)
@kotlin.internal.InlineOnly
public inline fun <T> MutableList<T>.remove(index: Int): T = removeAt(index)

```

\* Adds the specified [element] to this mutable collection.

```

@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(element: T) { this.add(element) }

```

\* Adds all elements of the given [elements] collection to this mutable collection.

```

@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Iterable<T>) { this.addAll(elements) }

```

\* Adds all elements of the given [elements] array to this mutable collection.

```

@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Array<T>) { this.addAll(elements) }

```

\* Adds all elements of the given [elements] sequence to this mutable collection.

```

@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Sequence<T>) { this.addAll(elements) }

```

\* Removes a single instance of the specified [element] from this mutable collection.

```

@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(element: T) { this.remove(element) }

```

\* Removes all elements contained in the given [elements] collection from this mutable collection.

```

@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Iterable<T>) { this.removeAll(elements) }

```

\* Removes all elements contained in the given [elements] array from this mutable collection.

```

@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Array<T>) { this.removeAll(elements) }

```

Removes all elements contained in the given [elements] sequence from this mutable collection.

```

@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Sequence<T>) { this.removeAll(elements) }

```

\* Adds all elements of the given [elements] collection to this [MutableCollection].

```

public fun <T> MutableCollection<in T>.addAll(elements: Iterable<T>): Boolean {
 when (elements) {
 is Collection -> return addAll(elements)
 else -> {
 var result: Boolean = false
 for (item in elements)
 if (add(item)) result = true
 return result
 }
 }
}

```

\* Adds all elements of the given [elements] sequence to this [MutableCollection].

```

public fun <T> MutableCollection<in T>.addAll(elements: Sequence<T>): Boolean {
 var result: Boolean = false
 for (item in elements)
 if (add(item)) result = true
 return result
}

```

\* Adds all elements of the given [elements] array to this [MutableCollection].

```

public fun <T> MutableCollection<in T>.addAll(elements: Array<out T>): Boolean {
 return addAll(elements.asList())
}

```

\* Removes all elements from this [MutableIterable] that match the given [predicate].

```

public fun <T> MutableIterable<T>.removeAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, true)

```

Retains only elements of this [MutableIterable] that match the given [predicate].

```

public fun <T> MutableIterable<T>.retainAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, false)

```

```

<T> MutableIterable<T>.filterInPlace(predicate: (T) -> Boolean, predicateResultToRemove: Boolean): Boolean
{
 var result = false
 with(iterator()) {
 while (hasNext())
 if (predicate(next()) ==
 predicateResultToRemove)
 remove()
 result = true
 }
 return result
}
* Removes all elements from this [MutableList] that match the given [predicate].

public fun <T> MutableList<T>.removeAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, true)
* Retains only elements of this [MutableList] that match the given [predicate].

public fun <T> MutableList<T>.retainAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, false)

private fun <T> MutableList<T>.filterInPlace(predicate: (T) -> Boolean, predicateResultToRemove: Boolean): Boolean {
 if (this !is RandomAccess)
 return (this as
 MutableIterable<T>).filterInPlace(predicate, predicateResultToRemove)
 var writeIndex: Int = 0
 for (readIndex in 0..lastIndex) {
 val element = this[readIndex]
 if (predicate(element) ==
 predicateResultToRemove)
 continue
 if (writeIndex != readIndex)
 this[writeIndex] =
 element
 writeIndex++
 }
 if (writeIndex < size)
 for (removeIndex in lastIndex
 downTo writeIndex)
 removeAt(removeIndex)
 return true
} else {
 return false
}
}
* Removes all elements from this [MutableCollection] that are also contained in the given
[elements] collection.

public fun <T> MutableCollection<in T>.removeAll(elements: Iterable<T>):
Boolean {
 return removeAll(elements.convertToSetForSetOperationWith(this))
}
* Removes all elements from this [MutableCollection] that are also contained in the given [elements] sequence.

public fun <T> MutableCollection<in T>.removeAll(elements: Sequence<T>): Boolean {
 val set =
 elements.toHashSet()
 return set.isNotEmpty() && removeAll(set)
}
* Removes all elements from this [MutableCollection] that are also contained in the given [elements] array.

public fun <T>
MutableCollection<in T>.removeAll(elements: Array<out T>): Boolean {
 return elements.isNotEmpty() &&
 removeAll(elements.toHashSet())
}
* Retains only elements of this [MutableCollection] that are
contained in the given [elements] collection.

public fun <T> MutableCollection<in T>.retainAll(elements:
Iterable<T>): Boolean {
 return retainAll(elements.convertToSetForSetOperationWith(this))
}
* Retains only elements of this [MutableCollection] that are contained in the given [elements] array.

public fun <T> MutableCollection<in T>.retainAll(elements: Array<out T>): Boolean {
 if
 (elements.isNotEmpty())
 return retainAll(elements.toHashSet())
 else
 return
 retainNothing()
}
* Retains only elements of this [MutableCollection] that are contained in the
given [elements] sequence.

public fun <T> MutableCollection<in T>.retainAll(elements: Sequence<T>):
Boolean {
 val set = elements.toHashSet()
 if (set.isNotEmpty())
 return retainAll(set)
 else
 return retainNothing()
}

private fun MutableCollection<*>.retainNothing(): Boolean {
 val result =
 isEmpty()
 clear()
 return result
}
* Randomly shuffles elements in this mutable list
using the specified [random] instance as the source of randomness.

* See:
https://en.wikipedia.org/wiki/Fisher%20%80%93Yates_shuffle#The_modern_algorithm

@SinceKotlin("1.3")
public fun <T> MutableList<T>.shuffle(random: Random): Unit {
 for (i in
 lastIndex downTo 1) {
 val j = random.nextInt(i + 1)
 val copy = this[i]
 this[i] = this[j]
 this[j] = copy
 }
}
* Returns a new list with the elements of this list randomly shuffled
using the specified [random] instance as the source of randomness.

@SinceKotlin("1.3")
public fun
<T> Iterable<T>.shuffled(random: Random): List<T> = toMutableList().apply {
 shuffle(random)
}

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")
package
kotlin.collections
private open class ReversedListReadOnly<out T>(private val delegate: List<T>) :
AbstractList<T>() {
 override val size: Int get() = delegate.size
 override fun get(index: Int): T =
 delegate[reverseElementIndex(index)]
}
private class ReversedList<T>(private val delegate:
MutableList<T>) : AbstractMutableList<T>() {
 override val size: Int get() = delegate.size
 override fun
 get(index: Int): T = delegate[reverseElementIndex(index)]
 override fun clear() = delegate.clear()
}

```

```
override fun removeAt(index: Int): T = delegate.removeAt(reverseElementIndex(index))\r\n\r\n override fun
set(index: Int, element: T): T = delegate.set(reverseElementIndex(index), element)\r\n override fun add(index: Int,
element: T) {\r\n delegate.add(reversePositionIndex(index), element)\r\n } \r\n}\r\n\r\nprivate fun
List<*>.reverseElementIndex(index: Int) =\r\n if (index in 0..lastIndex) lastIndex - index else throw
IndexOutOfBoundsException("Element index $index must be in range [0..lastIndex].")\r\n\r\nprivate fun
List<*>.reversePositionIndex(index: Int) =\r\n if (index in 0..size) size - index else throw
IndexOutOfBoundsException("Position index $index must be in range [0..size].")\r\n\r\n\r\n/*\r\n * Returns a
reversed read-only view of the original List.\r\n * All changes made in the original list will be reflected in the
reversed one.\r\n * @sample samples.collections.ReversedViews.asReversedList\r\n */\r\npublic fun <T>
List<T>.asReversed(): List<T> = ReversedListReadOnly(this)\r\n\r\n/*\r\n * Returns a reversed mutable view of
the original mutable List.\r\n * All changes made in the original list will be reflected in the reversed one and vice
versa.\r\n * @sample samples.collections.ReversedViews.asReversedMutableList\r\n */\r\n\r\n@kotlin.jvm.JvmName("asReversedMutable")\r\npublic fun <T> MutableList<T>.asReversed():
MutableList<T> = ReversedList(this)\r\n\r\n", /*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n\r\n
*\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("SequencesKt")\r\n@file:UseExperim
ental(ExperimentalTypeInference::class)\r\n\r\npackage kotlin.sequences\r\n\r\nimport kotlin.coroutines.*\r\nimport
kotlin.coroutines.intrinsics.*\r\nimport kotlin.experimental.ExperimentalTypeInference\r\n\r\n/*\r\n * Builds a
[Sequence] lazily yielding values one by one.\r\n * @see kotlin.sequences.generateSequence\r\n * @sample
samples.collections.Sequences.Building.buildSequenceYieldAll\r\n * @sample
samples.collections.Sequences.Building.buildFibonacciSequence\r\n * @SinceKotlin("1.3")\r\npublic fun <T>
sequence(@BuilderInference block: suspend SequenceScope<T>().() -> Unit): Sequence<T> = Sequence {
iterator(block) }\r\n\r\n@SinceKotlin("1.3")\r\n@Deprecated("Use 'sequence { }' function instead.",
ReplaceWith("sequence(builderAction)"), level =
DeprecationLevel.ERROR)\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T>
buildSequence(@BuilderInference noinline builderAction: suspend SequenceScope<T>().() -> Unit): Sequence<T> =
Sequence { iterator(builderAction) }\r\n\r\n/*\r\n * Builds an [Iterator] lazily yielding values one by one.\r\n * @sample
samples.collections.Sequences.Building.buildIterator\r\n * @sample
samples.collections.Iterables.Building.iterable\r\n * @SinceKotlin("1.3")\r\npublic fun <T>
iterator(@BuilderInference block: suspend SequenceScope<T>().() -> Unit): Iterator<T> {\r\n val iterator =
SequenceBuilderIterator<T>()\r\n iterator.nextStep = block.createCoroutineUnintercepted(receiver = iterator,
completion = iterator)\r\n return iterator\r\n}\r\n\r\n@SinceKotlin("1.3")\r\n@Deprecated("Use 'iterator { }'
function instead.", ReplaceWith("iterator(builderAction)"), level =
DeprecationLevel.ERROR)\r\n@kotlin.internal.InlineOnly\r\npublic inline fun <T>
buildIterator(@BuilderInference noinline builderAction: suspend SequenceScope<T>().() -> Unit): Iterator<T> =
iterator(builderAction)\r\n\r\n/*\r\n * The scope for yielding values of a [Sequence] or an [Iterator], provides
[yield] and [yieldAll] suspension functions.\r\n * @see sequence\r\n * @see iterator\r\n * @sample
samples.collections.Sequences.Building.buildSequenceYieldAll\r\n * @sample
samples.collections.Sequences.Building.buildFibonacciSequence\r\n\r\n
*\r\n@RestrictsSuspension\r\n@SinceKotlin("1.3")\r\npublic abstract class SequenceScope<in T> internal
constructor() {\r\n /*\r\n * Yields a value to the [Iterator] being built and suspends\r\n * until the next value
is requested.\r\n * @sample samples.collections.Sequences.Building.buildSequenceYieldAll\r\n *
*\r\n @sample samples.collections.Sequences.Building.buildFibonacciSequence\r\n * @public abstract suspend
fun yield(value: T)\r\n\r\n /*\r\n * Yields all values from the `iterator` to the [Iterator] being built\r\n * and
suspends until all these values are iterated and the next one is requested.\r\n * @sample
samples.collections.Sequences.Building.buildSequenceYieldAll\r\n * @public abstract suspend fun
```

```

yieldAll(iterator: Iterator<T>)\r\n\r\n /**\r\n * Yields a collections of values to the [Iterator] being built\r\n *
and suspends until all these values are iterated and the next one is requested.\r\n * \r\n * @sample
samples.collections.Sequences.Building.buildSequenceYieldAll\r\n * \r\n public suspend fun yieldAll(elements:
Iterable<T>) {\r\n if (elements is Collection && elements.isEmpty()) return\r\n return
yieldAll(elements.iterator())\r\n }\r\n\r\n /**\r\n * Yields potentially infinite sequence of values to the
[Iterator] being built\r\n * and suspends until all these values are iterated and the next one is requested.\r\n * \r\n
* The sequence can be potentially infinite.\r\n * \r\n * @sample
samples.collections.Sequences.Building.buildSequenceYieldAll\r\n * \r\n public suspend fun
yieldAll(sequence: Sequence<T>) = yieldAll(sequence.iterator())\r\n}\r\n\r\n@Deprecated("Use SequenceScope
class instead.", ReplaceWith("SequenceScope<T>"), level = DeprecationLevel.ERROR)\r\npublic typealias
SequenceBuilder<T> = SequenceScope<T>\r\n\r\nprivate typealias State = Int\r\n\r\nprivate const val
State_NotReady: State = 0\r\n\r\nprivate const val State_ManyNotReady: State = 1\r\n\r\nprivate const val
State_ManyReady: State = 2\r\n\r\nprivate const val State_Ready: State = 3\r\n\r\nprivate const val State_Done: State =
4\r\n\r\nprivate const val State_Failed: State = 5\r\n\r\n\r\nprivate class SequenceBuilderIterator<T> :
SequenceScope<T>(), Iterator<T>, Continuation<Unit> {\r\n private var state = State_NotReady\r\n private var
nextValue: T? = null\r\n private var nextIterator: Iterator<T>? = null\r\n var nextStep: Continuation<Unit>? =
null\r\n\r\n override fun hasNext(): Boolean {\r\n while (true) {\r\n when (state) {\r\n
State_NotReady -> {\r\n State_ManyNotReady ->\r\n if (nextIterator!!.hasNext()) {\r\n
state = State_ManyReady\r\n return true\r\n } else {\r\n nextIterator
= null\r\n }\r\n State_Done -> return false\r\n State_Ready, State_ManyReady ->
return true\r\n else -> throw exceptionalState()\r\n }\r\n\r\n state = State_Failed\r\n val
step = nextStep!!\r\n nextStep = null\r\n step.resume(Unit)\r\n }\r\n }\r\n\r\n override fun
next(): T {\r\n when (state) {\r\n State_NotReady, State_ManyNotReady -> return nextNotReady()\r\n
State_ManyReady -> {\r\n state = State_ManyNotReady\r\n return nextIterator!!.next()\r\n
}\r\n State_Ready -> {\r\n state = State_NotReady\r\n
}\r\n\r\n @Suppress("UNCHECKED_CAST")\r\n val result = nextValue as T\r\n nextValue = null\r\n
return result\r\n }\r\n else -> throw exceptionalState()\r\n }\r\n }\r\n\r\n private fun
nextNotReady(): T {\r\n if (!hasNext()) throw NoSuchElementException() else return next()\r\n }\r\n\r\n
private fun exceptionalState(): Throwable = when (state) {\r\n State_Done -> NoSuchElementException()\r\n
State_Failed -> IllegalStateException("Iterator has failed.")\r\n else -> IllegalStateException("Unexpected
state of the iterator: $state")\r\n }\r\n\r\n\r\n override suspend fun yield(value: T) {\r\n nextValue =
value\r\n state = State_Ready\r\n return suspendCoroutineUninterceptedOrReturn { c ->\r\n nextStep
= c\r\n COROUTINE_SUSPENDED\r\n }\r\n }\r\n\r\n override suspend fun yieldAll(iterator:
Iterator<T>) {\r\n if (iterator.hasNext()) return\r\n nextIterator = iterator\r\n state =
State_ManyReady\r\n return suspendCoroutineUninterceptedOrReturn { c ->\r\n nextStep = c\r\n
COROUTINE_SUSPENDED\r\n }\r\n }\r\n\r\n // Completion continuation implementation\r\n override
fun resumeWith(result: Result<Unit>) {\r\n result.getOrThrow() // just rethrow exception if it is there\r\n
state = State_Done\r\n }\r\n\r\n override val context: CoroutineContext\r\n get() =
EmptyCoroutineContext\r\n}\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.collections\r\n\r\ninternal fun checkWindowSizeStep(size: Int,
step: Int) {\r\n require(size > 0 && step > 0) {\r\n if (size != step)\r\n "Both size $size and step $step
must be greater than zero." \r\n else\r\n "size $size must be greater than zero." \r\n }\r\n }\r\n\r\n\r\ninternal
fun <T> Sequence<T>.windowedSequence(size: Int, step: Int, partialWindows: Boolean, reuseBuffer: Boolean):
Sequence<List<T>> {\r\n checkWindowSizeStep(size, step)\r\n return Sequence { windowedIterator(iterator(),
size, step, partialWindows, reuseBuffer) }\r\n}\r\n\r\n\r\ninternal fun <T> windowedIterator(iterator: Iterator<T>, size:
Int, step: Int, partialWindows: Boolean, reuseBuffer: Boolean): Iterator<List<T>> {\r\n if (!iterator.hasNext())
return EmptyIterator\r\n return iterator<List<T>> {\r\n val gap = step - size\r\n if (gap >= 0) {\r\n

```

```

var buffer = ArrayList<T>(size)\r\n var skip = 0\r\n for (e in iterator) {\r\n if (skip > 0) { skip
- = 1; continue }\r\n buffer.add(e)\r\n if (buffer.size == size) {\r\n yield(buffer)\r\n
 if (reuseBuffer) buffer.clear() else buffer = ArrayList(size)\r\n skip = gap\r\n }\r\n
}\r\n if (buffer.isNotEmpty()) {\r\n if (partialWindows || buffer.size == size) yield(buffer)\r\n
}\r\n } else {\r\n val buffer = RingBuffer<T>(size)\r\n for (e in iterator) {\r\n
buffer.add(e)\r\n if (buffer.isFull()) {\r\n yield(if (reuseBuffer) buffer else
ArrayList(buffer))\r\n buffer.removeFirst(step)\r\n }\r\n }\r\n if (partialWindows)
{\r\n while (buffer.size > step) {\r\n yield(if (reuseBuffer) buffer else ArrayList(buffer))\r\n
 buffer.removeFirst(step)\r\n }\r\n if (buffer.isNotEmpty()) yield(buffer)\r\n }\r\n
}\r\n }\r\n}\r\n\r\ninternal class MovingSubList<out E>(private val list: List<E>) : AbstractList<E>(),
RandomAccess {\r\n private var fromIndex: Int = 0\r\n private var _size: Int = 0\r\n\r\n fun move(fromIndex:
Int, toIndex: Int) {\r\n checkRangeIndexes(fromIndex, toIndex, list.size)\r\n this.fromIndex = fromIndex\r\n
 this._size = toIndex - fromIndex\r\n }\r\n\r\n override fun get(index: Int): E {\r\n
checkElementIndex(index, _size)\r\n return list[fromIndex + index]\r\n }\r\n\r\n override val size: Int
get() = _size\r\n}\r\n\r\n\r\n\r\n\r\n**\r\n * Provides ring buffer implementation.\r\n * Buffer overflow is not allowed
so [add] doesn't overwrite tail but raises an exception.\r\n * \r\nprivate class RingBuffer<T>(val capacity: Int) :
AbstractList<T>(), RandomAccess {\r\n init {\r\n require(capacity >= 0) { \"ring buffer capacity should not be
negative but it is $capacity\" }\r\n }\r\n\r\n private val buffer = arrayOfNulls<Any?>(capacity)\r\n private var
startIndex: Int = 0\r\n\r\n override var size: Int = 0\r\n private set\r\n\r\n override fun get(index: Int): T {\r\n
 checkElementIndex(index, size)\r\n @Suppress(\"UNCHECKED_CAST\")\r\n return
buffer[startIndex.forward(index)] as T\r\n }\r\n\r\n fun isFull() = size == capacity\r\n\r\n override fun
iterator(): Iterator<T> = object : AbstractIterator<T>() {\r\n private var count = size\r\n private var index =
startIndex\r\n\r\n override fun computeNext() {\r\n if (count == 0) {\r\n done()\r\n } else
{\r\n @Suppress(\"UNCHECKED_CAST\")\r\n setNext(buffer[index] as T)\r\n index =
index.forward(1)\r\n count--\r\n }\r\n }\r\n }\r\n}\r\n\r\n\r\n@Suppress(\"UNCHECKED_CAST\")\r\n override fun <T> toArray(array: Array<T>): Array<T> {\r\n val
result: Array<T?> =\r\n if (array.size < this.size) array.copyOf(this.size) else array as Array<T?>\r\n\r\n val size = this.size\r\n\r\n var widx = 0\r\n var idx = startIndex\r\n\r\n while (widx < size && idx <
capacity) {\r\n result[widx] = buffer[idx] as T\r\n widx++\r\n idx++\r\n }\r\n\r\n idx =
0\r\n while (widx < size) {\r\n result[widx] = buffer[idx] as T\r\n widx++\r\n idx++\r\n
}\r\n if (result.size > this.size) result[this.size] = null\r\n\r\n return result as Array<T>\r\n}\r\n\r\n\r\noverride fun toArray(): Array<Any?> {\r\n return toArray(arrayOfNulls(size))\r\n}\r\n\r\n\r\n/**\r\n * Add
[element] to the buffer or fail with [IllegalStateException] if no free space available in the buffer\r\n * \r\n fun
add(element: T) {\r\n if (isFull()) {\r\n throw IllegalStateException(\"ring buffer is full\")\r\n
}\r\n\r\n buffer[startIndex.forward(size)] = element\r\n size++\r\n }\r\n\r\n\r\n/**\r\n * Removes [n] first
elements from the buffer or fails with [IllegalArgumentException] if not enough elements in the buffer to
remove\r\n * \r\n fun removeFirst(n: Int) {\r\n require(n >= 0) { \"n shouldn't be negative but it is $n\" }\r\n
 require(n <= size) { \"n shouldn't be greater than the buffer size: n = $n, size = $size\" }\r\n\r\n if (n > 0)
{\r\n val start = startIndex\r\n val end = start.forward(n)\r\n\r\n if (start > end) {\r\n
buffer.fill(null, start, capacity)\r\n buffer.fill(null, 0, end)\r\n } else {\r\n buffer.fill(null,
start, end)\r\n }\r\n startIndex = end\r\n size -= n\r\n }\r\n }\r\n}\r\n\r\n\r\n@Suppress(\"NOTHING_TO_INLINE\")\r\n private inline fun Int.forward(n: Int): Int = (this + n) %
capacity\r\n}\r\n\r\n\r\n\r\n\r\n**\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n * \r\n\r\n\r\n\r\npackage kotlin.collections\r\n\r\n// UByteArray
===== \r\n\r\n@E
xperimentalUnsignedTypes\r\nprivate fun partition(\r\n array: UByteArray, left: Int, right: Int): Int {\r\n var i =
left\r\n var j = right\r\n val pivot = array[(left + right) / 2]\r\n while (i <= j) {\r\n while (array[i] <

```

```

pivot)\r\n i++\r\n while (array[j] > pivot)\r\n j--\r\n if (i <= j) {\r\n val tmp =
array[i]\r\n array[i] = array[j]\r\n array[j] = tmp\r\n i++\r\n j--\r\n }\r\n }\r\n
return i\r\n}\r\n\r\n@ExperimentalUnsignedTypes\r\nprivate fun quickSort(\r\n array: UByteArray, left: Int, right:
Int) {\r\n val index = partition(array, left, right)\r\n if (left < index - 1)\r\n quickSort(array, left, index - 1)\r\n if (index < right)\r\n quickSort(array, index, right)\r\n}\r\n\r\n// UShortArray
=====
\r\n@E
xperimentalUnsignedTypes\r\nprivate fun partition(\r\n array: UShortArray, left: Int, right: Int): Int {\r\n var i =
left\r\n var j = right\r\n val pivot = array[(left + right) / 2]\r\n while (i <= j) {\r\n while (array[i] <
pivot)\r\n i++\r\n while (array[j] > pivot)\r\n j--\r\n if (i <= j) {\r\n val tmp =
array[i]\r\n array[i] = array[j]\r\n array[j] = tmp\r\n i++\r\n j--\r\n }\r\n }\r\n
return i\r\n}\r\n\r\n@ExperimentalUnsignedTypes\r\nprivate fun quickSort(\r\n array: UShortArray, left: Int, right:
Int) {\r\n val index = partition(array, left, right)\r\n if (left < index - 1)\r\n quickSort(array, left, index - 1)\r\n if (index < right)\r\n quickSort(array, index, right)\r\n}\r\n\r\n// UIntArray
=====
\r\n@E
xperimentalUnsignedTypes\r\nprivate fun partition(\r\n array: UIntArray, left: Int, right: Int): Int {\r\n var i =
left\r\n var j = right\r\n val pivot = array[(left + right) / 2]\r\n while (i <= j) {\r\n while (array[i] <
pivot)\r\n i++\r\n while (array[j] > pivot)\r\n j--\r\n if (i <= j) {\r\n val tmp =
array[i]\r\n array[i] = array[j]\r\n array[j] = tmp\r\n i++\r\n j--\r\n }\r\n }\r\n
return i\r\n}\r\n\r\n@ExperimentalUnsignedTypes\r\nprivate fun quickSort(\r\n array: UIntArray, left: Int, right:
Int) {\r\n val index = partition(array, left, right)\r\n if (left < index - 1)\r\n quickSort(array, left, index - 1)\r\n if (index < right)\r\n quickSort(array, index, right)\r\n}\r\n\r\n// ULongArray
=====
\r\n@E
xperimentalUnsignedTypes\r\nprivate fun partition(\r\n array: ULongArray, left: Int, right: Int): Int {\r\n var i =
left\r\n var j = right\r\n val pivot = array[(left + right) / 2]\r\n while (i <= j) {\r\n while (array[i] <
pivot)\r\n i++\r\n while (array[j] > pivot)\r\n j--\r\n if (i <= j) {\r\n val tmp =
array[i]\r\n array[i] = array[j]\r\n array[j] = tmp\r\n i++\r\n j--\r\n }\r\n }\r\n
return i\r\n}\r\n\r\n@ExperimentalUnsignedTypes\r\nprivate fun quickSort(\r\n array: ULongArray, left: Int, right:
Int) {\r\n val index = partition(array, left, right)\r\n if (left < index - 1)\r\n quickSort(array, left, index - 1)\r\n if (index < right)\r\n quickSort(array, index, right)\r\n}\r\n\r\n\r\n// Interfaces
=====
\r\n/*
\r\n * Sorts the given array using qsort algorithm.\r\n */\r\n@ExperimentalUnsignedTypes\r\ninternal fun
sortArray(array: UByteArray) = quickSort(array, 0, array.size - 1)\r\n@ExperimentalUnsignedTypes\r\ninternal
fun sortArray(array: UShortArray) = quickSort(array, 0, array.size -
1)\r\n@ExperimentalUnsignedTypes\r\ninternal fun sortArray(array: UIntArray) = quickSort(array, 0, array.size -
1)\r\n@ExperimentalUnsignedTypes\r\ninternal fun sortArray(array: ULongArray) = quickSort(array, 0, array.size
- 1)", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of
this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*/\r\n\r\npackage kotlin.contracts\r\n\r\nimport kotlin.internal.ContractsDsl\r\nimport
kotlin.internal.InlineOnly\r\n\r\n/*\r\n * This marker distinguishes the experimental contract declaration API and is
used to opt-in for that feature\r\n * when declaring contracts of user functions.\r\n */\r\n * Any usage of a declaration
annotated with `@ExperimentalContracts` must be accepted either by\r\n * annotating that usage with the
[UseExperimental] annotation, e.g. `@UseExperimental(ExperimentalContracts::class)`,\r\n * or by using the
compiler argument `-Xuse-experimental=kotlin.contracts.ExperimentalContracts`.
\r\n
*/\r\n@Retention(AnnotationRetention.BINARY)\r\n@SinceKotlin("1.3")\r\n@Experimental\r\n@MustBeDocum
ented\r\npublic annotation class ExperimentalContracts\r\n\r\n/*\r\n * Provides a scope, where the functions of the
contract DSL, such as [returns], [callsInPlace], etc.,\r\n * can be used to describe the contract of a function.\r\n */\r\n
* This type is used as a receiver type of the lambda function passed to the [contract] function.\r\n */\r\n * @see
contract\r\n */\r\n@ContractsDsl\r\n@ExperimentalContracts\r\n@SinceKotlin("1.3")\r\npublic interface

```



```

ContractBuilder {
 /**
 * Describes a situation when a function returns normally, without any exceptions
 thrown.
 * Use [SimpleEffect.implies] function to describe a conditional effect that happens in such
 case.
 */
 @sample samples.contracts.returnsContract
 @ContractsDsl public fun returns():
Returns
 /**
 * Describes a situation when a function returns normally with the specified return
 [value].
 * The possible values of [value] are limited to `true`, `false` or `null`.
 * Use
 [SimpleEffect.implies] function to describe a conditional effect that happens in such case.
 */
 @sample samples.contracts.returnsTrueContract
 // @sample samples.contracts.returnsFalseContract
 //
 @sample samples.contracts.returnsNullContract
 @ContractsDsl public fun returns(value: Any?):
Returns
 /**
 * Describes a situation when a function returns normally with any value that is not
 `null`.
 * Use [SimpleEffect.implies] function to describe a conditional effect that happens in such
 case.
 */
 @sample samples.contracts.returnsNotNullContract
 @ContractsDsl public fun
returnsNotNull(): ReturnsNotNull
 /**
 * Specifies that the function parameter [lambda] is invoked in
 place.
 * This contract specifies that:
 * 1. the function [lambda] can only be invoked during the
 call of the owner function,
 * and it won't be invoked after that owner function call is completed;
 * 2.
 _ (optionally) _ the function [lambda] is invoked the amount of times specified by the [kind] parameter,
 * see
 the [InvocationKind] enum for possible values.
 * A function declaring the `callsInPlace` effect must
 be inline.
 */
 @sample samples.contracts.callsInPlaceAtMostOnceContract
 * @sample
samples.contracts.callsInPlaceAtLeastOnceContract
 * @sample
samples.contracts.callsInPlaceExactlyOnceContract
 * @sample
samples.contracts.callsInPlaceUnknownContract
 @ContractsDsl public fun <R> callsInPlace(lambda:
Function<R>, kind: InvocationKind = InvocationKind.UNKNOWN): CallsInPlace
 /**
 * Specifies
 how many times a function invokes its function parameter in place.
 * See [ContractBuilder.callsInPlace]
 for the details of the call-in-place function contract.
 */
 @ContractsDsl
 @ExperimentalContracts
 @SinceKotlin("1.3")
 public enum class InvocationKind
 {
 /**
 * A function parameter will be invoked one time or not invoked at all.
 */
 @sample
samples.contracts.callsInPlaceAtMostOnceContract
 @ContractsDsl AT_MOST_ONCE,
 /**
 *
 A function parameter will be invoked one or more times.
 */
 @sample
samples.contracts.callsInPlaceAtLeastOnceContract
 @ContractsDsl AT_LEAST_ONCE,
 /**
 *
 A function parameter will be invoked exactly one time.
 */
 @sample
samples.contracts.callsInPlaceExactlyOnceContract
 @ContractsDsl EXACTLY_ONCE,
 /**
 *
 A function parameter is called in place, but it's unknown how many times it can be called.
 */
 @sample
samples.contracts.callsInPlaceUnknownContract
 @ContractsDsl UNKNOWN
 }
 /**
 * Specifies the contract of a function.
 * The contract description must be at the beginning of a function and
 have at least one effect.
 * Only the top-level functions can have a contract for now.
 * @param
 builder the lambda where the contract of a function is described with the help of the [ContractBuilder] members.
 */
 @sample samples.contracts.returnsContract
 * @sample samples.contracts.returnsTrueContract
 *
 @sample samples.contracts.returnsFalseContract
 * @sample samples.contracts.returnsNullContract
 *
 @sample samples.contracts.returnsNotNullContract
 * @sample
samples.contracts.callsInPlaceAtMostOnceContract
 * @sample
samples.contracts.callsInPlaceAtLeastOnceContract
 * @sample
samples.contracts.callsInPlaceExactlyOnceContract
 * @sample
samples.contracts.callsInPlaceUnknownContract
 @ContractsDsl
 @ExperimentalContracts
 @InlineOn
ly
 @SinceKotlin("1.3")
 @Suppress("UNUSED_PARAMETER")
 public inline fun contract(builder:
ContractBuilder.() -> Unit) { }
 /**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 license/LICENSE.txt file.
 */
 @package kotlin.coroutines
 /**
 * Marks coroutine context element
 that intercepts coroutine continuations.
 * The coroutines framework uses [ContinuationInterceptor.Key] to
 retrieve the interceptor and
 * intercepts all coroutine continuations with [interceptContinuation] invocations.
 */

```

```

* Since Kotlin("1.3")
public interface ContinuationInterceptor : CoroutineContext.Element {
 * The key that defines the context interceptor.
 companion object Key :
 CoroutineContext.Key<ContinuationInterceptor>
 /** Returns continuation that wraps the original
 [continuation], thus intercepting all resumptions.
 * This function is invoked by coroutines framework when
 needed and the resulting continuations are
 * cached internally per each instance of the original
 [continuation].
 * This function may simply return original [continuation] if it does not want to
 intercept this particular continuation.
 * When the original [continuation] completes, coroutine
 framework invokes [releaseInterceptedContinuation]
 * with the resulting continuation if it was intercepted,
 that is if `interceptContinuation` had previously
 * returned a different continuation instance.
 */
 public fun <T> interceptContinuation(continuation: Continuation<T>): Continuation<T>
 /** Invoked for the continuation instance returned by [interceptContinuation] when the original
 continuation
 completes and will not be used anymore. This function is invoked only if [interceptContinuation]
 had
 returned a different continuation instance from the one it was invoked with.
 * Default implementation
 does nothing.
 * @param continuation Continuation instance returned by this interceptor's
 [interceptContinuation] invocation.
 */
 public fun releaseInterceptedContinuation(continuation:
 Continuation<*>) {
 /* do nothing by default */
 }
 // Performance optimization for a singleton
 Key
 public override operator fun <E : CoroutineContext.Element> get(key: CoroutineContext.Key<E>): E?
 = @Suppress("UNCHECKED_CAST")
 if (key === Key) this as E else null
 //
 Performance optimization to a singleton Key
 public override fun minusKey(key: CoroutineContext.Key<*>):
 CoroutineContext =
 if (key === Key) EmptyCoroutineContext else this
}
 /** Copyright 2010-
 2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by
 the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package
kotlin.coroutines
/** Persistent context for the coroutine. It is an indexed set of [Element] instances.
 * An indexed set is a mix between a set and a map.
 * Every element in this set has a unique [Key].
 */
* Since Kotlin("1.3")
public interface CoroutineContext {
 /** Returns the element with the
 given [key] from this context or `null`.
 */
 public operator fun <E : Element> get(key: Key<E>):
 E?
 /** Accumulates entries of this context starting with [initial] value and applying [operation]
 * from left to right to current accumulator value and each element of this context.
 */
 public fun <R>
 fold(initial: R, operation: (R, Element) -> R): R
 /** Returns a context containing elements from this
 context and elements from other [context].
 * The elements from this context with the same key as in the other
 one are dropped.
 */
 public operator fun plus(context: CoroutineContext): CoroutineContext =
 if (context === EmptyCoroutineContext) this else // fast path -- avoid lambda creation
 context.fold(this) {
 acc, element ->
 val removed = acc.minusKey(element.key)
 if (removed ===
 EmptyCoroutineContext) element else {
 // make sure interceptor is always last in the context (and
 thus is fast to get when present)
 val interceptor = removed[ContinuationInterceptor]
 if (interceptor == null) CombinedContext(removed, element) else {
 val left =
 removed.minusKey(ContinuationInterceptor)
 if (left === EmptyCoroutineContext)
 CombinedContext(CombinedContext(left,
 element), interceptor)
 else {
 CombinedContext(CombinedContext(left,
 element), interceptor)
 }
 }
 }
 }
 /** Returns a context containing
 elements from this context, but without an element with
 * the specified [key].
 */
 public fun
 minusKey(key: Key<*>): CoroutineContext
 /** Key for the elements of [CoroutineContext]. [E] is
 a type of element with this key.
 */
 public interface Key<E : Element>
 /** An element of
 the [CoroutineContext]. An element of the coroutine context is a singleton context by itself.
 */
 public interface Element : CoroutineContext {
 /** A key of this coroutine context element.
 */
 public val key: Key<*>
 public override operator fun <E : Element> get(key: Key<E>): E? =
 @Suppress("UNCHECKED_CAST")
 if (this.key == key) this as E else null
 public override
 fun <R> fold(initial: R, operation: (R, Element) -> R): R =
 operation(initial, this)
 public
 override fun minusKey(key: Key<*>): CoroutineContext =
 if (this.key == key) EmptyCoroutineContext
 }
}

```

```

else this } } \r\n", "/* \r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. \r\n */ \r\n \r\npackage kotlin.coroutines \r\n \r\nimport kotlin.coroutines.CoroutineContext. * \r\nimport kotlin.io.Serializable \r\n \r\n/** \r\n * Base class for [CoroutineContext.Element] implementations. \r\n * \r\n * @SinceKotlin("1.3") \r\n */ \r\npublic abstract class AbstractCoroutineContextElement(public override val key: Key<*>) : Element \r\n \r\n /** \r\n * An empty coroutine context. \r\n * \r\n * @SinceKotlin("1.3") \r\n */ \r\npublic object EmptyCoroutineContext : CoroutineContext, Serializable { \r\n private const val serialVersionUID: Long = 0 \r\n private fun readResolve(): Any = EmptyCoroutineContext \r\n public override fun <E : Element> get(key: Key<E>): E? = null \r\n public override fun <R> fold(initial: R, operation: (R, Element) -> R): R = initial \r\n public override fun plus(context: CoroutineContext): CoroutineContext = context \r\n public override fun minusKey(key: Key<*>): CoroutineContext = this \r\n public override fun hashCode(): Int = 0 \r\n public override fun toString(): String = "EmptyCoroutineContext" \r\n \r\n //----- internal impl ----- \r\n // this class is not exposed, but is hidden inside implementations \r\n // this is a left-biased list, so that `plus` works naturally \r\n @SinceKotlin("1.3") \r\n internal class CombinedContext(\r\n private val left: CoroutineContext, \r\n private val element: Element \r\n) : CoroutineContext, Serializable { \r\n override fun <E : Element> get(key: Key<E>): E? { \r\n var cur = this \r\n while (true) { \r\n cur.element[key]?.let { return it } \r\n val next = cur.left \r\n if (next is CombinedContext) { \r\n cur = next \r\n } else { \r\n return next[key] \r\n } \r\n } \r\n } \r\n public override fun <R> fold(initial: R, operation: (R, Element) -> R): R = \r\n operation(left.fold(initial, operation), element) \r\n public override fun minusKey(key: Key<*>): CoroutineContext { \r\n element[key]?.let { return left } \r\n val newLeft = left.minusKey(key) \r\n return when { \r\n newLeft === left -> this \r\n newLeft === EmptyCoroutineContext -> element \r\n else -> CombinedContext(newLeft, element) \r\n } \r\n } \r\n private fun size(): Int { \r\n var cur = this \r\n var size = 2 \r\n while (true) { \r\n cur = cur.left as? CombinedContext? \r\n size++ \r\n } \r\n } \r\n private fun contains(element: Element): Boolean = \r\n get(element.key) == element \r\n private fun containsAll(context: CombinedContext): Boolean { \r\n var cur = context \r\n while (true) { \r\n if (!contains(cur.element)) return false \r\n val next = cur.left \r\n if (next is CombinedContext) { \r\n cur = next \r\n } else { \r\n return contains(next as Element) \r\n } \r\n } \r\n } \r\n override fun equals(other: Any?): Boolean = \r\n this === other || other is CombinedContext && other.size() == size() && other.containsAll(this) \r\n override fun hashCode(): Int = left.hashCode() + element.hashCode() \r\n override fun toString(): String = \r\n "[" + fold("") { acc, element -> \r\n if (acc.isEmpty()) element.toString() else "$acc, $element" \r\n } + "]" \r\n private fun writeReplace(): Any { \r\n val n = size() \r\n val elements = arrayOfNulls<CoroutineContext>(n) \r\n var index = 0 \r\n fold(Unit) { _, element -> elements[index++] = element } \r\n check(index == n) \r\n @Suppress("UNCHECKED_CAST") \r\n return Serialized(elements as Array<CoroutineContext>) \r\n } \r\n private class Serialized(val elements: Array<CoroutineContext>) : Serializable { \r\n companion object { \r\n private const val serialVersionUID: Long = 0L \r\n } \r\n private fun readResolve(): Any = elements.fold(EmptyCoroutineContext, CoroutineContext::plus) \r\n } \r\n } \r\n \r\n", "/* \r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. \r\n */ \r\n \r\n@file:kotlin.jvm.JvmName("IntrinsicsKt") \r\n@file:kotlin.jvm.JvmMultifileClass \r\n \r\npackage kotlin.coroutines.intrinsics \r\n \r\nimport kotlin.coroutines. * \r\nimport kotlin.internal.InlineOnly \r\n \r\n /** \r\n * Obtains the current continuation instance inside suspend functions and either suspends \r\n * currently running coroutine or returns result immediately without suspension. \r\n * \r\n * If the [block] returns the special [COROUTINE_SUSPENDED] value, it means that suspend function did suspend the execution and will \r\n * not return any result immediately. In this case, the [Continuation] provided to the [block] shall be \r\n * resumed by invoking [Continuation.resumeWith] at some moment in the \r\n * future when the result becomes available to

```

resume the computation.  
 \* Otherwise, the return value of the [block] must have a type assignable to [T] and represents the result of this suspend function.  
 \* It means that the execution was not suspended and the [Continuation] provided to the [block] shall not be invoked.  
 \* As the result type of the [block] is declared as `Any?` and cannot be correctly type-checked, its proper return type remains on the conscience of the suspend function's author.  
 \* Invocation of [Continuation.resumeWith] resumes coroutine directly in the invoker's thread without going through the [ContinuationInterceptor] that might be present in the coroutine's [CoroutineContext].  
 \* It is the invoker's responsibility to ensure that a proper invocation context is established.  
 \* [Continuation.intercepted] can be used to acquire the intercepted continuation.  
 \* Note that it is not recommended to call either [Continuation.resume] nor [Continuation.resumeWithException] functions synchronously in the same stackframe where suspension function is run. Use [suspendCoroutine] as a safer way to obtain current continuation instance.

```

*\/@SinceKotlin("1.3")@InlineOnly@Suppress("UNUSED_PARAMETER",
 "RedundantSuspendModifier")public suspend inline fun <T>
suspendCoroutineUninterceptedOrReturn(crossinline block: (Continuation<T>) -> Any?): T = throw
NotImplementedError("Implementation of suspendCoroutineUninterceptedOrReturn is intrinsic")
This value is used as a return value of [suspendCoroutineUninterceptedOrReturn] `block` argument to state that
the execution was suspended and will not return any result immediately.
**Note: this value should not be used in general code.
** Using it outside of the context of `suspendCoroutineUninterceptedOrReturn`
function return value (including, but not limited to, storing this value in other properties, returning it from
other functions, etc) can lead to unspecified behavior of the code.
\/ It is implemented as property with
getter to avoid ProGuard <clinit> problem with multifile IntrinsicKt class
@SinceKotlin("1.3")public val
COROUTINE_SUSPENDED: Any get() = CoroutineSingletons.COROUTINE_SUSPENDED
Using enum
here ensures two important properties:
1. It makes SafeContinuation serializable with all kinds of serialization
frameworks (since all of them natively support enums)
2. It improves debugging experience, since you clearly
see toString() value of those objects and what package they come from
@SinceKotlin("1.3")@PublishedApi
// This class is Published API via serialized representation of SafeContinuation, don't rename/move
internal enum
class CoroutineSingletons { COROUTINE_SUSPENDED, UNDECIDED, RESUMED }
*/
* Copyright
2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
Use of this source code is
governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
*/
package
kotlin.experimental
/** Performs a bitwise AND operation between the two values.
*\/@SinceKotlin("1.1")@kotlin.internal.InlineOnlypublic inline infix fun Byte.and(other: Byte): Byte =
(this.toInt() and other.toInt()).toByte()
** Performs a bitwise OR operation between the two values.
*\/@SinceKotlin("1.1")@kotlin.internal.InlineOnlypublic inline infix fun Byte.or(other: Byte): Byte =
(this.toInt() or other.toInt()).toByte()
** Performs a bitwise XOR operation between the two values.
*\/@SinceKotlin("1.1")@kotlin.internal.InlineOnlypublic inline infix fun Byte.xor(other: Byte): Byte =
(this.toInt() xor other.toInt()).toByte()
** Inverts the bits in this value.
*\/@SinceKotlin("1.1")@kotlin.internal.InlineOnlypublic inline fun Byte.inv(): Byte =
(this.toInt().inv()).toByte()
** Performs a bitwise AND operation between the two values.
*\/@SinceKotlin("1.1")@kotlin.internal.InlineOnlypublic inline infix fun Short.and(other: Short): Short =
(this.toInt() and other.toInt()).toShort()
** Performs a bitwise OR operation between the two values.
*\/@SinceKotlin("1.1")@kotlin.internal.InlineOnlypublic inline infix fun Short.or(other: Short): Short =
(this.toInt() or other.toInt()).toShort()
** Performs a bitwise XOR operation between the two values.
*\/@SinceKotlin("1.1")@kotlin.internal.InlineOnlypublic inline infix fun Short.xor(other: Short): Short =
(this.toInt() xor other.toInt()).toShort()
** Inverts the bits in this value.
*\/@SinceKotlin("1.1")@kotlin.internal.InlineOnlypublic inline fun Short.inv(): Short =
(this.toInt().inv()).toShort()
*/
* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.
Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.
*/
package kotlin.experimental
** The experimental marker for type

```

inference augmenting annotations. Any usage of a declaration annotated with `@ExperimentalTypeInference` must be accepted either by annotating that usage with the `[UseExperimental]` annotation, e.g. `@UseExperimental(ExperimentalTypeInference::class)` or by using the compiler argument `-Xuse-experimental=kotlin.experimental.ExperimentalTypeInference`.

`@Experimental(level = Experimental.Level.ERROR)`

`@MustBeDocumented`

`@Retention(AnnotationRetention.BINARY)`

`@Target(AnnotationTarget.ANNOTATION_CLASS)`

Since Kotlin 1.3, `ExperimentalTypeInference` is a public annotation class.

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

`package kotlin.internal`

Specifies that the corresponding type should be ignored during type inference.

`@Target(AnnotationTarget.TYPE)`

`@Retention(AnnotationRetention.BINARY)`

internal annotation class `NoInfer`

Specifies that the constraint built for the type during type inference should be an equality one.

`@Target(AnnotationTarget.TYPE)`

`@Retention(AnnotationRetention.BINARY)`

internal annotation class `Exact`

Specifies that a corresponding member has the lowest priority in overload resolution.

`@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY)`

`@Retention(AnnotationRetention.BINARY)`

internal annotation class `LowPriorityInOverloadResolution`

Specifies that the corresponding member has the highest priority in overload resolution. Effectively this means that an extension annotated with this annotation will win in overload resolution over a member with the same signature.

`@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY)`

`@Retention(AnnotationRetention.BINARY)`

internal annotation class `HidesMembers`

The value of this type parameter should be mentioned in input types (argument types, receiver type or expected type).

`@Target(AnnotationTarget.TYPE_PARAMETER)`

`@Retention(AnnotationRetention.BINARY)`

internal annotation class `OnlyInputTypes`

Specifies that this function should not be called directly without inlining.

`@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY, AnnotationTarget.PROPERTY_GETTER, AnnotationTarget.PROPERTY_SETTER)`

`@Retention(AnnotationRetention.BINARY)`

internal annotation class `InlineOnly`

Specifies that this declaration can have dynamic receiver type.

`@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY)`

`@Retention(AnnotationRetention.BINARY)`

internal annotation class `DynamicExtension`

The value of this parameter should be a property reference expression (`this::foo`), referencing a `lateinit` property, the backing field of which is accessible at the point where the corresponding argument is passed.

`@Target(AnnotationTarget.VALUE_PARAMETER)`

`@Retention(AnnotationRetention.BINARY)`

`@SinceKotlin("1.2")`

internal annotation class `AccessibleLateinitPropertyLiteral`

Specifies that this declaration is only completely supported since the specified version.

The Kotlin compiler of an earlier version is going to report a diagnostic on usages of this declaration. The diagnostic message can be specified with `[message]`, or via `[errorCode]` (takes less space, but might not be immediately clear to the user). The diagnostic severity can be specified with `[level]`: `WARNING`/`ERROR` mean that either a warning or an error is going to be reported, `HIDDEN` means that the declaration is going to be removed from resolution completely.

`[versionKind]` specifies which version should be compared with the `[version]` value, when compiling the usage of the annotated declaration. Note that prior to 1.2, only `[RequireKotlinVersionKind.LANGUAGE_VERSION]` was supported, so the Kotlin compiler before 1.2 is going to treat any `[RequireKotlin]` as if it requires the language version. Since 1.2, the Kotlin compiler supports `[RequireKotlinVersionKind.LANGUAGE_VERSION]`, `[RequireKotlinVersionKind.COMPILER_VERSION]` and `[RequireKotlinVersionKind.API_VERSION]`. If the actual value of `[versionKind]` is something different (e.g. a new version kind, added in future versions of Kotlin), Kotlin 1.2 is going to ignore this `[RequireKotlin]` altogether, whereas Kotlin before 1.2 is going to treat this as a requirement on the language version.

This annotation is erased at compile time; its arguments are stored in a more compact form in the Kotlin metadata.

```

@Target(AnnotationTarget.CLASS, AnnotationTarget.FUNCTION,
AnnotationTarget.PROPERTY, AnnotationTarget.CONSTRUCTOR,
AnnotationTarget.TYPEALIAS)@Retention(AnnotationRetention.SOURCE)@Repeatable@SinceKotlin(
"1.2")internal annotation class RequireKotlin(val version: String, val message: String = "",
val level: DeprecationLevel = DeprecationLevel.ERROR, val versionKind: RequireKotlinVersionKind =
RequireKotlinVersionKind.LANGUAGE_VERSION, val errorCode: Int = -1)internal enum class
RequireKotlinVersionKind {LANGUAGE_VERSION, COMPILER_VERSION,
API_VERSION} Specifies that this declaration is a part of special DSL, used for constructing
function's contract.
@Retention(AnnotationRetention.BINARY)@SinceKotlin("1.2")internal
annotation class ContractsDsl, /* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.
package kotlin.properties
import kotlin.reflect.KProperty
 * Standard property delegates.
 * Returns a property delegate for a
read/write property with a non-`null` value that is initialized not during
 * object construction time but at a later
time. Trying to read the property before the initial value has been
 * assigned results in an exception.
 * @sample samples.properties.Delegates.notNullDelegate
public fun <T : Any> notNull():
ReadWriteProperty<Any?, T> = NotNullVar()
 * Returns a property delegate for a read/write
property that calls a specified callback function when changed.
 * @param initialValue the initial value of the
property.
 * @param onChange the callback which is called after the change of the property is made. The value
of the property
 * has already been changed when this callback is invoked.
 * @sample
samples.properties.Delegates.observableDelegate
public inline fun <T> observable(initialValue: T,
crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Unit):
ReadWriteProperty<Any?, T> =
object : ObservableProperty<T>(initialValue) {
override fun
afterChange(property: KProperty<*>, oldValue: T, newValue: T) = onChange(property, oldValue, newValue)
}
 * Returns a property delegate for a read/write property that calls a specified callback function
when changed,
 * allowing the callback to veto the modification.
 * @param initialValue the initial value
of the property.
 * @param onChange the callback which is called before a change to the property value is
attempted.
 * The value of the property hasn't been changed yet, when this callback is invoked.
 * If the
callback returns `true` the value of the property is being set to the new value,
 * and if the callback returns
`false` the new value is discarded and the property remains its old value.
 * @sample
samples.properties.Delegates.vetoableDelegate
 * @sample
samples.properties.Delegates.throwVetoableDelegate
public inline fun <T> vetoable(initialValue: T,
crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Boolean):
ReadWriteProperty<Any?, T> =
object : ObservableProperty<T>(initialValue) {
override fun
beforeChange(property: KProperty<*>, oldValue: T, newValue: T): Boolean = onChange(property, oldValue,
newValue)
}
private class NotNullVar<T : Any>(): ReadWriteProperty<Any?, T> {
private var value: T? = null
public override fun getValue(thisRef: Any?, property: KProperty<*>): T {
return value ?: throw IllegalStateException("Property ${property.name} should be initialized before get.")
}
public override fun setValue(thisRef: Any?, property: KProperty<*>, value: T) {
this.value =
value
}
}
/* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.
package kotlin.properties
import kotlin.reflect.KProperty
 * Implements the core logic of a property delegate for a read/write property that calls callback functions when
changed.
 * @param initialValue the initial value of the property.
 * @public abstract class
ObservableProperty<T>(initialValue: T) : ReadWriteProperty<Any?, T> {
private var value =
initialValue
 * The callback which is called before a change to the property value is attempted.

```

```

 * The value of the property hasn't been changed yet, when this callback is invoked.
 * If the callback returns `true` the value of the property is being set to the new value,
 * and if the callback returns `false` the new value is discarded and the property remains its old value.
 * protected open fun beforeChange(property: KProperty<*>, oldValue: T, newValue: T): Boolean = true
 /**
 * The callback which is called after the change of the property is made. The value of the property
 * has already been changed when this callback is invoked.
 * protected open fun afterChange(property: KProperty<*>, oldValue: T, newValue: T): Unit
 {
 }
 public override fun getValue(thisRef: Any?, property: KProperty<*>): T {
 return value
 }
 public override fun setValue(thisRef: Any?, property: KProperty<*>, value: T) {
 val oldValue = this.value
 if (!beforeChange(property, oldValue, value)) {
 return
 }
 this.value = value
 afterChange(property, oldValue, value)
 }
}
"
"/
* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
package kotlin.random
import kotlin.math.nextDown
/**
 * An abstract class that is implemented by random number generator algorithms.
 * The companion object [Random.Default] is the default instance of [Random].
 * To get a seeded instance of random generator use [Random] function.
 * @sample samples.random.Randoms.defaultRandom
 * @SinceKotlin("1.3")
 */
public abstract class Random {
 /**
 * Gets the next random [bitCount] number of bits.
 * Generates an `Int` whose lower [bitCount] bits are filled with random values and the remaining upper bits are zero.
 * @param bitCount number of bits to generate, must be in range 0..32, otherwise the behavior is unspecified.
 */
 public abstract fun nextBits(bitCount: Int): Int
 /**
 * Gets the next random `Int` from the random number generator.
 * Generates an `Int` random value uniformly distributed between `Int.MIN_VALUE` and `Int.MAX_VALUE` (inclusive).
 */
 public open fun nextInt(): Int = nextBits(32)
 /**
 * Gets the next random non-negative `Int` from the random number generator less than the specified [until] bound.
 * Generates an `Int` random value uniformly distributed between `0` (inclusive) and the specified [until] bound (exclusive).
 * @param until must be positive.
 * @throws IllegalArgumentException if [until] is negative or zero.
 */
 public open fun nextInt(until: Int): Int = nextInt(0, until)
 /**
 * Gets the next random `Int` from the random number generator in the specified range.
 * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.
 * @throws IllegalArgumentException if [from] is greater than or equal to [until].
 */
 public open fun nextInt(from: Int, until: Int): Int {
 checkRangeBounds(from, until)
 val n = until - from
 if (n > 0 || n == Int.MIN_VALUE) {
 val rnd = if (n and -n == n) {
 val bitCount = fastLog2(n)
 nextBits(bitCount)
 } else {
 var v: Int
 do {
 val bits = nextInt().ushr(1)
 v = bits % n
 } while (bits - v + (n - 1) < 0)
 v
 }
 return from + rnd
 } else {
 while (true) {
 val rnd = nextInt()
 if (rnd in from until until) return rnd
 }
 }
 }
 /**
 * Gets the next random `Long` from the random number generator.
 * Generates a `Long` random value uniformly distributed between `Long.MIN_VALUE` and `Long.MAX_VALUE` (inclusive).
 */
 public open fun nextLong(): Long = nextInt().toLong().shl(32) + nextInt()
 /**
 * Gets the next random non-negative `Long` from the random number generator less than the specified [until] bound.
 * Generates a `Long` random value uniformly distributed between `0` (inclusive) and the specified [until] bound (exclusive).
 * @param until must be positive.
 * @throws IllegalArgumentException if [until] is negative or zero.
 */
 public open fun nextLong(until: Long): Long = nextLong(0, until)
 /**
 * Gets the next random `Long` from the random number generator in the specified range.
 * Generates a `Long` random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.
 * @throws IllegalArgumentException if [from] is greater than or equal to [until].
 */
 public open fun nextLong(from: Long, until: Long): Long {
 checkRangeBounds(from, until)
 val n = until - from
 if (n > 0) {
 val rnd: Long
 if (n and -n == n) {
 val nLow = n.toInt()
 val nHigh = (n ushr 32).toInt()
 }
 }
 }
}

```

```

rnd = when {
 nLow != 0 -> {
 val bitCount = fastLog2(nLow)
 toUInt().toLong()
 nextBits(bitCount).toLong() and 0xFFFF_FFFF
 }
 nHigh == 1 -> {
 // toUInt().toLong()
 nextInt().toLong() and 0xFFFF_FFFF
 }
 else -> {
 val bitCount = fastLog2(nHigh)
 nextBits(bitCount).toLong().shl(32) + nextInt()
 }
} else {
 var v: Long
 do {
 val bits = nextLong().ushr(1)
 v = bits % n
 }
 while (bits - v + (n - 1) < 0)
 rnd = v
 return from + rnd
} else {
 while (true) {
 val rnd = nextLong()
 if (rnd in from until until) return rnd
 }
}

/**
 * Gets the next random [Boolean] value.
 */
public open fun nextBoolean(): Boolean = nextBits(1) != 0

/**
 * Gets the next random [Double] value uniformly distributed between 0 (inclusive) and 1 (exclusive).
 */
public open fun nextDouble(): Double = doubleFromParts(nextBits(26), nextBits(27))

/**
 * Gets the next random non-negative `Double` from the random number generator less than the specified [until] bound.
 */
public open fun nextDouble(until: Double): Double = nextDouble(0.0, until)

/**
 * Gets the next random `Double` from the random number generator in the specified range.
 */
public open fun nextDouble(from: Double, until: Double): Double {
 checkRangeBounds(from, until)
 val size = until - from
 val r = if (size.isInfinite() && from.isFinite() && until.isFinite()) {
 val r1 = nextDouble() * (until / 2 - from / 2)
 from + r1 + r1
 } else {
 from + nextDouble() * size
 }
 return if (r >= until) until.nextDown() else r
}

/**
 * Gets the next random [Float] value uniformly distributed between 0 (inclusive) and 1 (exclusive).
 */
public open fun nextFloat(): Float = nextBits(24) / (1 shl 24).toFloat()

/**
 * Fills a subrange of the specified byte [array] starting from [fromIndex] inclusive and ending [toIndex] exclusive with random bytes.
 */
public open fun nextBytes(array: ByteArray, fromIndex: Int = 0, toIndex: Int = array.size): ByteArray {
 require(fromIndex in 0..array.size && toIndex in 0..array.size) {
 "\"fromIndex ($fromIndex) or toIndex ($toIndex) are out of range: 0..${array.size}\"
 }
 require(fromIndex <= toIndex) {
 "\"fromIndex ($fromIndex) must be not greater than toIndex ($toIndex).\""
 }
 val steps = (toIndex - fromIndex) / 4
 var position = fromIndex
 repeat(steps) {
 val v = nextInt()
 array[position] = v.toByte()
 array[position + 1] = v.ushr(8).toByte()
 array[position + 2] = v.ushr(16).toByte()
 array[position + 3] = v.ushr(24).toByte()
 position += 4
 }
 val remainder = toIndex - position
 val vr = nextBits(remainder * 8)
 for (i in 0 until remainder) {
 array[position + i] = vr.ushr(i * 8).toByte()
 }
 return array
}

/**
 * Fills the specified byte [array] with random bytes and returns it.
 */
public open fun nextBytes(array: ByteArray): ByteArray = nextBytes(array, 0, array.size)

/**
 * Creates a byte array of the specified [size], filled with random bytes.
 */
public open fun nextBytes(size: Int): ByteArray = nextBytes(ByteArray(size))

/**
 * The default random number generator.
 */
public open fun nextLong(): Long = defaultRandom.nextLong()

/**
 * On JVM this generator is thread-safe, its methods can be invoked from multiple threads.
 */
public open fun nextLong(until: Long): Long = defaultRandom.nextLong(until)

/**
 * On JVM this generator is thread-safe, its methods can be invoked from multiple threads.
 */
public open fun nextLong(from: Long, until: Long): Long = defaultRandom.nextLong(from, until)

/**
 * On JVM this generator is thread-safe, its methods can be invoked from multiple threads.
 */
public open fun nextBoolean(): Boolean =

```





```

[Int.MIN_VALUE] and [Int.MAX_VALUE] (inclusive).\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun Random.nextInt(): UInt =
nextInt().toUInt()\r\n\r\n**\r\n * Gets the next random [UInt] from the random number generator less than the
specified [until] bound.\r\n * Generates a [UInt] random value uniformly distributed between `0` (inclusive)
and the specified [until] bound (exclusive).\r\n * @throws IllegalArgumentException if [until] is zero.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun Random.nextInt(until: UInt): UInt =
nextInt(0u, until)\r\n\r\n**\r\n * Gets the next random [UInt] from the random number generator in the specified
range.\r\n * Generates a [UInt] random value uniformly distributed between the specified [from] (inclusive)
and [until] (exclusive) bounds.\r\n * @throws IllegalArgumentException if [from] is greater than or equal to
[until].\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun Random.nextInt(from:
UInt, until: UInt): UInt {\r\n checkUIntRangeBounds(from, until)\r\n\r\n val signedFrom = from.toInt() xor
Int.MIN_VALUE\r\n val signedUntil = until.toInt() xor Int.MIN_VALUE\r\n\r\n val signedResult =
nextInt(signedFrom, signedUntil) xor Int.MIN_VALUE\r\n return signedResult.toUInt()\r\n}\r\n\r\n**\r\n * Gets
the next random [UInt] from the random number generator in the specified [range].\r\n * Generates a [UInt]
random value uniformly distributed in the specified [range]:\r\n * from `range.start` inclusive to
`range.endInclusive` inclusive.\r\n * @throws IllegalArgumentException if [range] is empty.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun Random.nextInt(range: UIntRange):
UInt = when {\r\n range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range:
$range")\r\n range.last < UInt.MAX_VALUE -> nextUInt(range.first, range.last + 1u)\r\n range.first >
UInt.MIN_VALUE -> nextUInt(range.first - 1u, range.last) + 1u\r\n else -> nextUInt()\r\n}\r\n\r\n**\r\n * Gets
the next random [ULong] from the random number generator.\r\n * Generates a [ULong] random value
uniformly distributed between [ULong.MIN_VALUE] and [ULong.MAX_VALUE] (inclusive).\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun Random.nextULong(): ULong =
nextLong().toULong()\r\n\r\n**\r\n * Gets the next random [ULong] from the random number generator less than
the specified [until] bound.\r\n * Generates a [ULong] random value uniformly distributed between `0`
(inclusive) and the specified [until] bound (exclusive).\r\n * @throws IllegalArgumentException if [until] is
zero.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun Random.nextULong(until:
ULong): ULong = nextULong(0uL, until)\r\n\r\n**\r\n * Gets the next random [ULong] from the random number
generator in the specified range.\r\n * Generates a [ULong] random value uniformly distributed between the
specified [from] (inclusive) and [until] (exclusive) bounds.\r\n * @throws IllegalArgumentException if [from]
is greater than or equal to [until].\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun
Random.nextULong(from: ULong, until: ULong): ULong {\r\n checkULongRangeBounds(from, until)\r\n\r\n val
signedFrom = from.toLong() xor Long.MIN_VALUE\r\n val signedUntil = until.toLong() xor
Long.MIN_VALUE\r\n\r\n val signedResult = nextLong(signedFrom, signedUntil) xor Long.MIN_VALUE\r\n
return signedResult.toULong()\r\n}\r\n\r\n**\r\n * Gets the next random [ULong] from the random number
generator in the specified [range].\r\n * Generates a [ULong] random value uniformly distributed in the
specified [range]:\r\n * from `range.start` inclusive to `range.endInclusive` inclusive.\r\n * @throws
IllegalArgumentException if [range] is empty.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun Random.nextULong(range:
ULongRange): ULong = when {\r\n range.isEmpty() -> throw IllegalArgumentException("Cannot get random in
empty range: $range")\r\n range.last < ULong.MAX_VALUE -> nextULong(range.first, range.last + 1u)\r\n
range.first > ULong.MIN_VALUE -> nextULong(range.first - 1u, range.last) + 1u\r\n else ->
nextULong()\r\n}\r\n\r\n**\r\n * Fills the specified unsigned byte [array] with random bytes and returns it.\r\n
* @return [array] filled with random bytes.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun Random.nextUBytes(array:
UByteArray): UByteArray {\r\n nextBytes(array.asByteArray())\r\n return array\r\n}\r\n\r\n**\r\n * Creates an
unsigned byte array of the specified [size], filled with random bytes.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun Random.nextUBytes(size: Int):

```

```

ByteArray = nextBytes(size).asByteArray()\r\n\r\n**\r\n * Fills a subrange of the specified `UByte` [array]
starting from [fromIndex] inclusive and ending [toIndex] exclusive with random UBytes.\r\n * \r\n * @return [array]
with the subrange filled with random bytes.\r\n
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic fun Random.nextUBytes(array:
ByteArray, fromIndex: Int = 0, toIndex: Int = array.size): UByteArray {\r\n nextBytes(array.asByteArray()),
fromIndex, toIndex)\r\n return array\r\n}\r\n\r\n\r\n@ExperimentalUnsignedTypes\r\ninternal fun
checkUIntRangeBounds(from: UInt, until: UInt) = require(until > from) { boundsErrorMessage(from, until)
}\r\n@ExperimentalUnsignedTypes\r\ninternal fun checkULongRangeBounds(from: ULong, until: ULong) =
require(until > from) { boundsErrorMessage(from, until) }\r\n", "*/\r\n * Copyright 2010-2018 JetBrains s.r.o. and
Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that
can be found in the license/LICENSE.txt file.\r\n */\r\n\r\npackage kotlin.random\r\n\r\n**\r\n * Random number
generator, algorithm "xorwow" from p. 5 of Marsaglia, "Xorshift RNGs".\r\n * \r\n * Cycles after 2^160 * (2^32-
1) repetitions.\r\n * \r\n * See http://www.jstatsoft.org/v08/i14/paper for details.\r\n */\r\ninternal class
XorWowRandom\r\ninternal constructor(\r\n private var x: Int, \r\n private var y: Int, \r\n private var z: Int, \r\n
private var w: Int, \r\n private var v: Int, \r\n private var addend: Int\r\n) : Random() {\r\n\r\n internal
constructor(seed1: Int, seed2: Int) : \r\n this(seed1, seed2, 0, 0, seed1.inv(), (seed1 shl 10) xor (seed2 ushr
4))\r\n\r\n init {\r\n require((x or y or z or w or v) != 0) { "Initial state must have at least one non-zero
element." }\r\n\r\n // some trivial seeds can produce several values with zeroes in upper bits, so we discard first
64\r\n repeat(64) { nextInt() }\r\n }\r\n\r\n override fun nextInt(): Int {\r\n var t = x\r\n t = t xor (t
ushr 2)\r\n x = y\r\n y = z\r\n z = w\r\n val v0 = v\r\n w = v0\r\n t = (t xor (t shl 1)) xor v0
xor (v0 shl 4)\r\n v = t\r\n addend += 362437\r\n return t + addend\r\n }\r\n\r\n override fun
nextBits(bitCount: Int): Int = \r\n nextInt().takeUpperBits(bitCount)\r\n}\r\n", "*/\r\n * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("RangesKt")\r\n\r\npackage
kotlin.ranges\r\n\r\n**\r\n * Represents a range of floating point numbers.\r\n * Extends [ClosedRange] interface
providing custom operation [lessThanOrEquals] for comparing values of range domain type.\r\n * \r\n * This
interface is implemented by floating point ranges returned by [Float.rangeTo] and [Double.rangeTo] operators to\r\n
* achieve IEEE-754 comparison order instead of total order of floating point numbers.\r\n
*\r\n@SinceKotlin("1.1")\r\npublic interface ClosedFloatingPointRange<T> : Comparable<T>> :
ClosedRange<T> {\r\n override fun contains(value: T): Boolean = lessThanOrEquals(start, value) &&
lessThanOrEquals(value, endInclusive)\r\n override fun isEmpty(): Boolean = !lessThanOrEquals(start,
endInclusive)\r\n\r\n /**\r\n * Compares two values of range domain type and returns true if first is less than or
equal to second.\r\n */\r\n fun lessThanOrEquals(a: T, b: T): Boolean\r\n}\r\n\r\n**\r\n * Represents a range of
[Comparable] values.\r\n */\r\nprivate open class ComparableRange<T> : Comparable<T>>(\r\n override val start:
T, \r\n override val endInclusive: T\r\n) : ClosedRange<T> {\r\n\r\n override fun equals(other: Any?): Boolean
{\r\n return other is ComparableRange<*> && (isEmpty() && other.isEmpty()) || \r\n start ==
other.start && endInclusive == other.endInclusive)\r\n }\r\n\r\n override fun hashCode(): Int {\r\n return if
(isEmpty()) -1 else 31 * start.hashCode() + endInclusive.hashCode()\r\n }\r\n\r\n override fun toString(): String
= "$start..$endInclusive"\r\n}\r\n\r\n**\r\n * A closed range of values of type `Double`.\r\n * \r\n * Numbers are
compared with the ends of this range according to IEEE-754.\r\n */\r\nprivate class ClosedDoubleRange(\r\n start:
Double, \r\n endInclusive: Double\r\n) : ClosedFloatingPointRange<Double> {\r\n private val _start = start\r\n
private val _endInclusive = endInclusive\r\n override val start: Double get() = _start\r\n override val
endInclusive: Double get() = _endInclusive\r\n\r\n override fun lessThanOrEquals(a: Double, b: Double): Boolean
= a <= b\r\n\r\n override fun contains(value: Double): Boolean = value >= _start && value <= _endInclusive\r\n
\r\n override fun isEmpty(): Boolean = !(_start <= _endInclusive)\r\n\r\n override fun equals(other: Any?): Boolean
{\r\n return other is ClosedDoubleRange && (isEmpty() && other.isEmpty()) || \r\n _start ==
other._start && _endInclusive == other._endInclusive)\r\n }\r\n\r\n override fun hashCode(): Int {\r\n return

```



```

String = "\", marginPrefix: String = "\"): String {
 require(marginPrefix.isNotBlank()) { "marginPrefix must
 be non-blank string." }
 val lines = lines()
 return lines.reindent(length + newIndent.length * lines.size,
 getIndentFunction(newIndent), { line ->
 val firstNonWhitespaceIndex = line.indexOfFirst {
 !it.isWhitespace() }
 when {
 firstNonWhitespaceIndex == -1 -> null
 line.startsWith(marginPrefix, firstNonWhitespaceIndex) -> line.substring(firstNonWhitespaceIndex +
 marginPrefix.length)
 else -> null
 }
 })
}

Detects a common minimal indent
of all the input lines, removes it from every line and also removes the first and the last lines if they are blank
(notice difference blank vs empty).
Note that blank lines do not affect the detected indent level.
In case if there are non-blank lines with no leading whitespace characters (no indent at all) then the common
indent is 0, and therefore this function doesn't change the indentation.
Doesn't preserve the original line
endings.
@sample samples.text.Strings.trimIndent
@see trimMargin
kotlin.text.isBlank
public fun String.trimIndent(): String = replaceIndent("\")

Detects a
common minimal indent like it does [trimIndent] and replaces it with the specified [newIndent].
public fun
String.replaceIndent(newIndent: String = "\"): String {
 val lines = lines()
 val minCommonIndent =
 lines.filter(String::isNotBlank).map(String::indentWidth).min(): 0
 return
 lines.reindent(length + newIndent.length * lines.size, getIndentFunction(newIndent), { line ->
 line.drop(minCommonIndent) })
}

Prepends [indent] to every line of the original string.
Doesn't preserve the original line endings.
public fun String.prependIndent(indent: String = " "): String
= lineSequence().map {
 when {
 it.isBlank() ->
 when {
 it.length < indent.length -> indent
 else -> it
 }
 else -> indent + it
 }
}.joinToString("\n")

private fun String.indentWidth(): Int =
indexOfFirst { !it.isWhitespace() }.let { if (it == -1) length else it }

private fun getIndentFunction(indent:
String) = when {
 indent.isEmpty() -> { line: String -> line }
 else -> { line: String -> indent + line
}

private inline fun List<String>.reindent(resultSizeEstimate: Int, indentAddFunction: (String)
-> String, indentCutFunction: (String) -> String?): String {
 val lastIndex = lastIndex
 return
 mapIndexedNotNull { index, value ->
 if ((index == 0 || index == lastIndex) && value.isBlank())
 null
 else
 indentCutFunction(value)?.let(indentAddFunction)?: value
 }.joinTo(StringBuilder(resultSizeEstimate), "\n").toString()
}

Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.
Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.
package kotlin.text

Represents a collection of captured groups in a single match of a regular expression.
This collection has
size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.
Groups are
indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
An element
of the collection at the particular index can be `null` if the corresponding group in the regular expression is
optional and there was no match captured by that group.
public interface MatchGroupCollection :
Collection<MatchGroup?> {
 /** Returns a group with the specified [index].
 * @return An
 instance of [MatchGroup] if the group with the specified [index] was matched or `null` otherwise.
 Groups are indexed from 1 to the count of groups in the regular expression. A group with the index 0
 corresponds to the entire match.
 public operator fun get(index: Int): MatchGroup?
}

Extends [MatchGroupCollection] by introducing a way to get matched groups by name, when regex supports it.
@SinceKotlin("1.1")
public interface MatchNamedGroupCollection : MatchGroupCollection {
 /** Returns a named group with the specified [name].
 * @return An instance of [MatchGroup] if the
 group with the specified [name] was matched or `null` otherwise.
 * @throws IllegalArgumentException if
 there is no group with the specified [name] defined in the regex pattern.
 * @throws
 UnsupportedOperationException if getting named groups isn't supported on the current platform.
 public operator fun get(name: String): MatchGroup?
}

Represents the results from a single regular
expression match.
public interface MatchResult {
 /** The range of indices in the original string where
 match was captured.
 public val range: IntRange
 /** The substring from the input string captured by this

```

```

match. */\r\n public val value: String\r\n /**\r\n * A collection of groups matched by the regular
expression.\r\n */\r\n * This collection has size of `groupCount + 1` where `groupCount` is the count of groups
in the regular expression.\r\n * Groups are indexed from 1 to `groupCount` and group with the index 0
corresponds to the entire match.\r\n */\r\n public val groups: MatchGroupCollection\r\n /**\r\n * A list of
matched indexed group values.\r\n */\r\n * This list has size of `groupCount + 1` where `groupCount` is the
count of groups in the regular expression.\r\n * Groups are indexed from 1 to `groupCount` and group with the
index 0 corresponds to the entire match.\r\n */\r\n * If the group in the regular expression is optional and there
were no match captured by that group,\r\n * corresponding item in [groupValues] is an empty string.\r\n */\r\n
* @sample samples.text.Regexps.matchDeconstructingToGroupValues\r\n */\r\n public val groupValues:
List<String>\r\n\r\n /**\r\n * An instance of [MatchResult.Deconstructed] wrapper providing components for
destructuring assignment of group values.\r\n */\r\n * component1 corresponds to the value of the first group,
component2 \u0024 of the second, and so on.\r\n */\r\n * @sample
samples.text.Regexps.matchDeconstructingToGroupValues\r\n */\r\n public val destructured: Deconstructed get()
= Deconstructed(this)\r\n\r\n /** Returns a new [MatchResult] with the results for the next match, starting at the
position\r\n * at which the last match ended (at the character after the last matched character).\r\n */\r\n\r\n
public fun next(): MatchResult?\r\n\r\n /**\r\n * Provides components for destructuring assignment of group
values.\r\n */\r\n * [component1] corresponds to the value of the first group, [component2] \u0024 of the
second, and so on.\r\n */\r\n * If the group in the regular expression is optional and there were no match
captured by that group,\r\n * corresponding component value is an empty string.\r\n */\r\n * @sample
samples.text.Regexps.matchDeconstructingToGroupValues\r\n */\r\n public class Deconstructed internal
constructor(public val match: MatchResult) {\r\n @kotlin.internal.InlineOnly\r\n public operator inline fun
component1(): String = match.groupValues[1]\r\n @kotlin.internal.InlineOnly\r\n public operator inline
fun component2(): String = match.groupValues[2]\r\n @kotlin.internal.InlineOnly\r\n public operator
inline fun component3(): String = match.groupValues[3]\r\n @kotlin.internal.InlineOnly\r\n public
operator inline fun component4(): String = match.groupValues[4]\r\n @kotlin.internal.InlineOnly\r\n
public operator inline fun component5(): String = match.groupValues[5]\r\n @kotlin.internal.InlineOnly\r\n
public operator inline fun component6(): String = match.groupValues[6]\r\n @kotlin.internal.InlineOnly\r\n
public operator inline fun component7(): String = match.groupValues[7]\r\n @kotlin.internal.InlineOnly\r\n
public operator inline fun component8(): String = match.groupValues[8]\r\n @kotlin.internal.InlineOnly\r\n
public operator inline fun component9(): String = match.groupValues[9]\r\n @kotlin.internal.InlineOnly\r\n
public operator inline fun component10(): String = match.groupValues[10]\r\n\r\n /**\r\n * Returns
destructured group values as a list of strings.\r\n * First value in the returned list corresponds to the value of
the first group, and so on.\r\n */\r\n * @sample
samples.text.Regexps.matchDeconstructingToGroupValues\r\n */\r\n public fun toList(): List<String> =
match.groupValues.subList(1, match.groupValues.size)\r\n } \r\n }"/\r\n * Copyright 2010-2018 JetBrains s.r.o.
and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license
that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n package kotlin.text\r\n\r\n /**\r\n * Defines names for
Unicode symbols used in proper Typography.\r\n */\r\n\r\n public object Typography {\r\n /** The character "
\u0024 quotation mark */\r\n public const val quote: Char = "\u0022"\r\n /** The character $ \u0024 dollar
sign */\r\n public const val dollar: Char = "\u0024"\r\n /** The character & \u0024 ampersand */\r\n
public const val amp: Char = "\u0026"\r\n /** The character < \u0024 less-than sign */\r\n public const val
less: Char = "\u003C"\r\n /** The character > \u0024 greater-than sign */\r\n public const val greater: Char
= "\u003E"\r\n /** The non-breaking space character */\r\n public const val nbsp: Char = "\u00A0"\r\n /** The
character × */\r\n public const val times: Char = "\u00D7"\r\n /** The character ¢ */\r\n public
const val cent: Char = "\u00A2"\r\n /** The character £ */\r\n public const val pound: Char = "\u00A3"\r\n
/** The character § */\r\n public const val section: Char = "\u00A7"\r\n /** The character © */\r\n
public const val copyright: Char = "\u00A9"\r\n /** The character « */\r\n public const val
leftGuillemete: Char = "\u00AB"\r\n /** The character » */\r\n public const val rightGuillemete: Char =

```

```

\u00BB\r\n /** The character ® */\r\n public const val registered: Char = "\u00AE"\r\n /** The
character ° */\r\n public const val degree: Char = "\u00B0"\r\n /** The character ± */\r\n public
const val plusMinus: Char = "\u00B1"\r\n /** The character ¶ */\r\n public const val paragraph: Char =
\u00B6\r\n /** The character · */\r\n public const val middleDot: Char = "\u00B7"\r\n /** The
character ½ */\r\n public const val half: Char = "\u00BD"\r\n /** The character – */\r\n public
const val ndash: Char = "\u2013"\r\n /** The character — */\r\n public const val mdash: Char =
\u2014\r\n /** The character ‘ */\r\n public const val leftSingleQuote: Char = "\u2018"\r\n /** The
character ’ */\r\n public const val rightSingleQuote: Char = "\u2019"\r\n /** The character ‚
*/\r\n public const val lowSingleQuote: Char = "\u201A"\r\n /** The character “ */\r\n public const
val leftDoubleQuote: Char = "\u201C"\r\n /** The character ” */\r\n public const val rightDoubleQuote:
Char = "\u201D"\r\n /** The character „ */\r\n public const val lowDoubleQuote: Char = "\u201E"\r\n
/** The character † */\r\n public const val dagger: Char = "\u2020"\r\n /** The character ‡
*/\r\n public const val doubleDagger: Char = "\u2021"\r\n /** The character • */\r\n public const val
bullet: Char = "\u2022"\r\n /** The character … */\r\n public const val ellipsis: Char = "\u2026"\r\n /**
The character ′ */\r\n public const val prime: Char = "\u2032"\r\n /** The character ″ */\r\n
public const val doublePrime: Char = "\u2033"\r\n /** The character € */\r\n public const val euro:
Char = "\u20AC"\r\n /** The character ™ */\r\n public const val tm: Char = "\u2122"\r\n /** The
character ≈ */\r\n public const val almostEqual: Char = "\u2248"\r\n /** The character ≠ */\r\n
public const val notEqual: Char = "\u2260"\r\n /** The character ≤ */\r\n public const val lessOrEqual:
Char = "\u2264"\r\n /** The character ≥ */\r\n public const val greaterOrEqual: Char =
\u2265\r\n}"/\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n *
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*/\r\n\r\npackage kotlin.time\r\n\r\n/**\r\n * A source of time for measuring time intervals.\r\n */\r\n * The only
operation provided by the clock is [markNow]. It returns a [ClockMark], which can be used to query the elapsed
time later.\r\n */\r\n * @see [measureTime]\r\n * @see [measureTimedValue]\r\n
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalTime\r\npublic interface Clock {\r\n /**\r\n * Marks a time point
on this clock.\r\n */\r\n * The returned [ClockMark] instance encapsulates captured time point and allows
querying\r\n * the duration of time interval [elapsed][ClockMark.elapsedNow] from that point.\r\n */\r\n
public fun markNow(): ClockMark\r\n}\r\n\r\n/**\r\n * Represents a time point notched on a particular [Clock].
Remains bound to the clock it was taken from\r\n * and allows querying for the duration of time elapsed from that
point (see the function [elapsedNow]).\r\n */\r\n@SinceKotlin("1.3")\r\n@ExperimentalTime\r\npublic abstract
class ClockMark {\r\n /**\r\n * Returns the amount of time passed from this clock mark on the clock from
which this mark was taken.\r\n */\r\n * Note that the value returned by this function can change on subsequent
invocations.\r\n */\r\n public abstract fun elapsedNow(): Duration\r\n\r\n /**\r\n * Returns a clock mark on
the same clock that is ahead of this clock mark by the specified [duration].\r\n */\r\n * The returned clock mark
is more _late_ when the [duration] is positive, and more _early_ when the [duration] is negative.\r\n
*/\r\n public open operator fun plus(duration: Duration): ClockMark = AdjustedClockMark(this, duration)\r\n\r\n /**\r\n
* Returns a clock mark on the same clock that is behind this clock mark by the specified [duration].\r\n */\r\n
* The returned clock mark is more _early_ when the [duration] is positive, and more _late_ when the [duration] is
negative.\r\n */\r\n public open operator fun minus(duration: Duration): ClockMark = plus(-duration)\r\n\r\n\r\n
/**\r\n * Returns true if this clock mark has passed according to the clock from which this mark was taken.\r\n
*/\r\n * Note that the value returned by this function can change on subsequent invocations.\r\n * If the clock is
monotonic, it can change only from false to true, namely, when the clock mark becomes behind the current point
of the clock.\r\n */\r\n public fun hasPassedNow(): Boolean = !elapsedNow().isNegative()\r\n\r\n /**\r\n *
Returns false if this clock mark has not passed according to the clock from which this mark was taken.\r\n */\r\n
* Note that the value returned by this function can change on subsequent invocations.\r\n * If the clock is
monotonic, it can change only from true to false, namely, when the clock mark becomes behind the current point
of the clock.\r\n */\r\n public fun hasNotPassedNow(): Boolean =

```

```

elapsedNow().isNegative()\r\n}\r\n\r\n\r\n\r\n@ExperimentalTime\r\n\r\n@SinceKotlin("1.3")\r\n\r\n@kotlin.internal.Inline
Only\r\n\r\n@Deprecated("Subtracting one ClockMark from another is not a well defined operation because these
clock marks could have been obtained from the different clocks.", level = DeprecationLevel.ERROR)\r\n\r\npublic
inline operator fun ClockMark.minus(other: ClockMark): Duration = throw Error("Operation is
disallowed.")\r\n\r\n\r\n\r\n@ExperimentalTime\r\n\r\n@SinceKotlin("1.3")\r\n\r\n@kotlin.internal.InlineOnly\r\n\r\n@Deprecated
("Comparing one ClockMark to another is not a well defined operation because these clock marks could have been
obtained from the different clocks.", level = DeprecationLevel.ERROR)\r\n\r\npublic inline operator fun
ClockMark.compareTo(other: ClockMark): Int = throw Error("Operation is
disallowed.")\r\n\r\n\r\n\r\n@ExperimentalTime\r\n\r\nprivate class AdjustedClockMark(val mark: ClockMark, val
adjustment: Duration) : ClockMark() {\r\n override fun elapsedNow(): Duration = mark.elapsedNow() -
adjustment\r\n\r\n override fun plus(duration: Duration): ClockMark = AdjustedClockMark(mark, adjustment +
duration)\r\n}\r\n"/*\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n *
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
/\r\n\r\npackage kotlin.time\r\n\r\n/\r\n * The most precise clock available in the platform.\r\n * \r\n * The clock
returns its readings from a source of monotonic time when it is available in a target platform,\r\n * and resorts to a
non-monotonic time source otherwise.\r\n */\r\n\r\n@SinceKotlin("1.3")\r\n\r\n@ExperimentalTime\r\n\r\npublic expect
object MonoClock : Clock\r\n\r\n/*\r\n * An abstract class used to implement clocks that return their readings as
[Long] values in the specified [unit].\r\n * \r\n * @property unit The unit in which this clock readings are
expressed.\r\n */\r\n\r\n@SinceKotlin("1.3")\r\n\r\n@ExperimentalTime\r\n\r\npublic abstract class
AbstractLongClock(protected val unit: DurationUnit) : Clock {\r\n /*\r\n * This protected method should be
overridden to return the current reading of the clock expressed as a [Long] number\r\n * in the unit specified by
the [unit] property.\r\n */\r\n protected abstract fun read(): Long\r\n\r\n private class LongClockMark(private
val startedAt: Long, private val clock: AbstractLongClock, private val offset: Duration) : ClockMark() {\r\n
 override fun elapsedNow(): Duration = (clock.read() - startedAt).toDuration(clock.unit) - offset\r\n
 override fun
plus(duration: Duration): ClockMark = LongClockMark(startedAt, clock, offset + duration)\r\n }\r\n\r\n override
fun markNow(): ClockMark = LongClockMark(read(), this, Duration.ZERO)\r\n}\r\n\r\n/*\r\n * An abstract class
used to implement clocks that return their readings as [Double] values in the specified [unit].\r\n * \r\n * @property
unit The unit in which this clock readings are expressed.\r\n */\r\n\r\n@SinceKotlin("1.3")\r\n\r\n@ExperimentalTime\r\n\r\npublic abstract class AbstractDoubleClock(protected val
unit: DurationUnit) : Clock {\r\n /*\r\n * This protected method should be overridden to return the current
reading of the clock expressed as a [Double] number\r\n * in the unit specified by the [unit] property.\r\n */\r\n
protected abstract fun read(): Double\r\n\r\n private class DoubleClockMark(private val startedAt: Double, private
val clock: AbstractDoubleClock, private val offset: Duration) : ClockMark() {\r\n override fun elapsedNow():
Duration = (clock.read() - startedAt).toDuration(clock.unit) - offset\r\n override fun plus(duration: Duration):
ClockMark = DoubleClockMark(startedAt, clock, offset + duration)\r\n }\r\n\r\n override fun markNow():
ClockMark = DoubleClockMark(read(), this, Duration.ZERO)\r\n}\r\n\r\n/*\r\n * A clock that has
programmatically updatable readings. It is useful as a predictable source of time in tests.\r\n * \r\n * The current
clock reading value can be advanced by the specified duration amount with the operator [plusAssign]:\r\n * \r\n *
```\r\n * val clock = TestClock()\r\n * clock += 10.seconds\r\n * ```\r\n * \r\n * Implementation note: the current
clock reading value is stored as a [Long] number of nanoseconds,\r\n * thus it's capable to represent a time range of
approximately 1292 years.\r\n * \r\n * Should the reading value overflow as the result of [plusAssign] operation, an
[IllegalStateException] is thrown.\r\n */\r\n\r\n@SinceKotlin("1.3")\r\n\r\n@ExperimentalTime\r\n\r\npublic class TestClock
: AbstractLongClock(unit = DurationUnit.NANOSECONDS) {\r\n    private var reading: Long = 0L\r\n\r\n    override fun
read(): Long = reading\r\n\r\n    /*\r\n     * Advances the current reading value of this clock by the
specified [duration].\r\n     * \r\n     * [duration] value is rounded down towards zero when converting it to a [Long]
number of nanoseconds.\r\n     * \r\n     * For example, if the duration being added is `0.6.nanoseconds`, the clock
reading won't advance because\r\n     * the duration value will be rounded to zero nanoseconds.\r\n     */\r\n    @throws
IllegalStateException when the reading value overflows as the result of this operation.\r\n    public operator

```



```

fun plusAssign(duration: Duration) {
    val delta = duration.toDouble(unit)
    val longDelta = delta.toLong()
    reading = if (longDelta != Long.MIN_VALUE && longDelta != Long.MAX_VALUE) {
        // when delta fits in long, add it as long
        val newReading = reading + longDelta
        if (reading xor longDelta >= 0 && reading xor newReading < 0) overflow(duration)
        newReading
    } else {
        // when delta is greater than long, add it as double
        val newReading = reading + delta
        if (newReading > Long.MAX_VALUE || newReading < Long.MIN_VALUE) overflow(duration)
        newReading.toDouble()
    }
}

private fun overflow(duration: Duration) {
    throw IllegalStateException("TestClock will overflow if its reading ${reading}ns is advanced by $duration.")
}

public interface WallClock {
    fun currentTimeMilliseconds(): Long
}

companion object : WallClock, AbstractLongClock(unit = DurationUnit.MILLISECONDS) {
    override fun currentTimeMilliseconds(): Long = System.currentTimeMillis()
    override fun read(): Long = System.currentTimeMillis()
    override fun toString(): String = "WallClock(System.currentTimeMillis())"
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

import kotlin.time.*

@UseExperimental(ExperimentalTime::class)
private inline val storageUnit get() = DurationUnit.NANOSECONDS

/* Represents the amount of time one instant of time is away from another instant.
 * A negative duration is possible in a situation when the second instant is earlier than the first one.
 * An infinite duration value [Duration.INFINITE] can be used to represent infinite timeouts.
 * To construct a duration use either the extension function [toDuration], or the extension properties [hours], [minutes], [seconds], and so on, available on [Int], [Long], and [Double] numeric types.
 * To get the value of this duration expressed in a particular [duration units][DurationUnit] use the functions [toInt], [toLong], and [toDouble] or the properties [inHours], [inMinutes], [inSeconds], [inNanoseconds], and so on.
 */

@SinceKotlin("1.3")
@ExperimentalTime
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTORS_OF_INLINE_CLASS")
public inline class Duration internal constructor(internal val value: Double) : Comparable<Duration> {
    // TODO: backend fails on init block, wait for KT-28055
    init {
        require(!_value.isNaN().not())
    }

    companion object {
        /** The duration equal to exactly 0 seconds. */
        public val ZERO: Duration = Duration(0.0)

        /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
        public val INFINITE: Duration = Duration(Double.POSITIVE_INFINITY)

        /** Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified [targetUnit]. */
        public fun convert(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double = convertDurationUnit(value, sourceUnit, targetUnit)

        /** Returns the negative of this value. */
        public operator fun unaryMinus(): Duration = Duration(-value)

        /** Returns a duration whose value is the sum of this and [other] duration values. */
        public operator fun plus(other: Duration): Duration = Duration(value + other.value)

        /** Returns a duration whose value is the difference between this and [other] duration values. */
        public operator fun minus(other: Duration): Duration = Duration(value - other.value)

        /** Returns a duration whose value is this duration value multiplied by the given [scale] number. */
        public operator fun times(scale: Int): Duration = Duration(value * scale)

        /** Returns a duration whose value is this duration value multiplied by the given [scale] number. */
        public operator fun times(scale: Double): Duration = Duration(value * scale)

        /** Returns a duration whose value is this duration value divided by the given [scale] number. */
        public operator fun div(scale: Int): Duration = Duration(value / scale)

        /** Returns a duration whose value is this duration value divided by the given [scale] number. */
        public operator fun div(scale: Double): Duration = Duration(value / scale)

        /** Returns a number that is the ratio of [this] and [other] duration values. */
        public operator fun div(other: Duration): Double = this.value / other.value

        /** Returns true, if the duration value is less than zero. */
        public fun isNegative(): Boolean = value < 0

        /** Returns true, if the duration value is greater than zero. */
        public fun isPositive(): Boolean = value >

```

```

0\r\n\r\n /** Returns true, if the duration value is infinite. */\r\n public fun isInfinite(): Boolean =
value.isInfinite()\r\n\r\n /** Returns true, if the duration value is finite. */\r\n public fun isFinite(): Boolean =
value.isFinite()\r\n\r\n /** Returns the absolute value of this value. The returned value is always non-negative.
*/\r\n public val absoluteValue: Duration get() = if (isNegative()) -this else this\r\n\r\n override fun
compareTo(other: Duration): Int = this.value.compareTo(other.value)\r\n\r\n\r\n // splitting to components\r\n\r\n
/**\r\n * Splits this duration into days, hours, minutes, seconds, and nanoseconds and executes the given [action]
with these components.\r\n * The result of [action] is returned as the result of this function.\r\n * -
`nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than
1_000_000_000;\r\n * - `seconds` represents the whole number of seconds in this duration, and its absolute value
is less than 60;\r\n * - `minutes` represents the whole number of minutes in this duration, and its absolute value
is less than 60;\r\n * - `hours` represents the whole number of hours in this duration, and its absolute value is less
than 24;\r\n * - `days` represents the whole number of days in this duration.\r\n * If the value doesn't fit in
[Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],\r\n * it is coerced into that
range.\r\n */\r\n public inline fun <T> toComponents(action: (days: Int, hours: Int, minutes: Int, seconds: Int,
nanoseconds: Int) -> T): T =\r\n    action(inDays.toInt(), hoursComponent, minutesComponent,
secondsComponent, nanosecondsComponent)\r\n\r\n /**\r\n * Splits this duration into hours, minutes, seconds,
and nanoseconds and executes the given [action] with these components.\r\n * The result of [action] is returned as
the result of this function.\r\n * - `nanoseconds` represents the whole number of nanoseconds in this
duration, and its absolute value is less than 1_000_000_000;\r\n * - `seconds` represents the whole number of
seconds in this duration, and its absolute value is less than 60;\r\n * - `minutes` represents the whole number of
minutes in this duration, and its absolute value is less than 60;\r\n * - `hours` represents the whole number of
hours in this duration.\r\n * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less
than [Int.MIN_VALUE],\r\n * it is coerced into that range.\r\n */\r\n public inline fun <T>
toComponents(action: (hours: Int, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T =\r\n
    action(inHours.toInt(), minutesComponent, secondsComponent, nanosecondsComponent)\r\n\r\n /**\r\n *
Splits this duration into minutes, seconds, and nanoseconds and executes the given [action] with these
components.\r\n * The result of [action] is returned as the result of this function.\r\n * - `nanoseconds`
represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;\r\n
* - `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;\r\n
* - `minutes` represents the whole number of minutes in this duration.\r\n * If the value doesn't fit in [Int] range,
i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],\r\n * it is coerced into that range.\r\n
*/\r\n public inline fun <T> toComponents(action: (minutes: Int, seconds: Int, nanoseconds: Int) -> T): T =\r\n
    action(inMinutes.toInt(), secondsComponent, nanosecondsComponent)\r\n\r\n /**\r\n * Splits this duration into
seconds, and nanoseconds and executes the given [action] with these components.\r\n * The result of [action] is
returned as the result of this function.\r\n * - `nanoseconds` represents the whole number of nanoseconds
in this duration, and its absolute value is less than 1_000_000_000;\r\n * - `seconds` represents the whole number
of seconds in this duration.\r\n * If the value doesn't fit in [Long] range, i.e. it's greater than
[Long.MAX_VALUE] or less than [Long.MIN_VALUE],\r\n * it is coerced into that range.\r\n */\r\n public
inline fun <T> toComponents(action: (seconds: Long, nanoseconds: Int) -> T): T =\r\n
    action(inSeconds.toLong(), nanosecondsComponent)\r\n\r\n @PublishedApi\r\n internal val hoursComponent:
Int get() = (inHours % 24).toInt()\r\n @PublishedApi\r\n internal val minutesComponent: Int get() = (inMinutes
% 60).toInt()\r\n @PublishedApi\r\n internal val secondsComponent: Int get() = (inSeconds % 60).toInt()\r\n
@PublishedApi\r\n internal val nanosecondsComponent: Int get() = (inNanoseconds % 1e9).toInt()\r\n\r\n\r\n //
conversion to units\r\n\r\n /** Returns the value of this duration expressed as a [Double] number of the specified
[unit]. */\r\n public fun toDouble(unit: DurationUnit): Double = convertDurationUnit(value, storageUnit,
unit)\r\n\r\n /**\r\n * Returns the value of this duration expressed as a [Long] number of the specified [unit].\r\n
*\r\n * If the value doesn't fit in the range of [Long] type, it is coerced into that range, see the conversion
[Double.toLong] for details.\r\n */\r\n public fun toLong(unit: DurationUnit): Long =

```

```

toDouble(unit).toLong()\r\n\r\n  /**\r\n   * Returns the value of this duration expressed as an [Int] number of the
specified [unit].\r\n   *\r\n   * If the value doesn't fit in the range of [Int] type, it is coerced into that range, see the
conversion [Double.toInt] for details.\r\n   */\r\n   public fun toInt(unit: DurationUnit): Int =
toDouble(unit).toInt()\r\n\r\n  /** The value of this duration expressed as a [Double] number of days. */\r\n
public val inDays: Double get() = toDouble(DurationUnit.DAYS)\r\n\r\n  /** The value of this duration expressed
as a [Double] number of hours. */\r\n   public val inHours: Double get() = toDouble(DurationUnit.HOURS)\r\n\r\n
/** The value of this duration expressed as a [Double] number of minutes. */\r\n   public val inMinutes: Double
get() = toDouble(DurationUnit.MINUTES)\r\n\r\n  /** The value of this duration expressed as a [Double] number
of seconds. */\r\n   public val inSeconds: Double get() = toDouble(DurationUnit.SECONDS)\r\n\r\n  /** The value
of this duration expressed as a [Double] number of milliseconds. */\r\n   public val inMilliseconds: Double get() =
toDouble(DurationUnit.MILLISECONDS)\r\n\r\n  /** The value of this duration expressed as a [Double] number
of microseconds. */\r\n   public val inMicroseconds: Double get() =
toDouble(DurationUnit.MICROSECONDS)\r\n\r\n  /** The value of this duration expressed as a [Double] number
of nanoseconds. */\r\n   public val inNanoseconds: Double get() =
toDouble(DurationUnit.NANOSECONDS)\r\n\r\n  // shortcuts\r\n\r\n  /**\r\n   * Returns the value of this
duration expressed as a [Long] number of nanoseconds.\r\n   *\r\n   * If the value doesn't fit in the range of [Long]
type, it is coerced into that range, see the conversion [Double.toLong] for details.\r\n   *\r\n   * The range of
durations that can be expressed as a `Long` number of nanoseconds is approximately \u00c2\u00b11292 years.\r\n
*/\r\n   public fun toLongNanoseconds(): Long = toLong(DurationUnit.NANOSECONDS)\r\n\r\n  /**\r\n   *
Returns the value of this duration expressed as a [Long] number of milliseconds.\r\n   *\r\n   * The value is
coerced to the range of [Long] type, if it doesn't fit in that range, see the conversion [Double.toLong] for details.\r\n
*/\r\n   * The range of durations that can be expressed as a `Long` number of milliseconds is approximately
\u00c2\u00b11292 million years.\r\n   */\r\n   public fun toLongMilliseconds(): Long =
toLong(DurationUnit.MILLISECONDS)\r\n\r\n  /**\r\n   * Returns a string representation of this duration value
expressed in the unit which yields the most compact and readable number value.\r\n   *\r\n   * Special cases:\r\n
* - zero duration is formatted as `"0s"`\r\n   * - the infinite duration is formatted as `"Infinity"` without unit\r\n
* - very small durations (less than 1e-15 s) are expressed in seconds and formatted in scientific notation\r\n   * -
very big durations (more than 1e+7 days) are expressed in days and formatted in scientific notation\r\n   *\r\n
* @return the value of duration in the automatically determined unit followed by that unit abbreviated name: `d`, `h`,
`m`, `s`, `ms`, `us`, or `ns`.\r\n   *\r\n   * @sample samples.time.Durations.toStringDefault\r\n   */\r\n
fun toString(): String = when {\r\n   isInfinite() -> value.toString()\r\n   value == 0.0 -> "0s"\r\n   else ->
{\r\n   val absNs = absoluteValue.inNanoseconds\r\n   var scientific = false\r\n   var maxDecimals =
0\r\n   val unit = when {\r\n   absNs < 1e-6 -> DurationUnit.SECONDS.also { scientific = true }\r\n
absNs < 1 -> DurationUnit.NANOSECONDS.also { maxDecimals = 7 }\r\n   absNs < 1e3 ->
DurationUnit.NANOSECONDS\r\n   absNs < 1e6 -> DurationUnit.MICROSECONDS\r\n   absNs
< 1e9 -> DurationUnit.MILLISECONDS\r\n   absNs < 1000e9 -> DurationUnit.SECONDS\r\n
absNs < 60_000e9 -> DurationUnit.MINUTES\r\n   absNs < 3600_000e9 -> DurationUnit.HOURS\r\n
absNs < 86400e9 * 1e7 -> DurationUnit.DAYS\r\n   else -> DurationUnit.DAYS.also { scientific = true
}\r\n   }\r\n   val value = toDouble(unit)\r\n   when {\r\n   scientific ->
formatScientific(value)\r\n   maxDecimals > 0 -> formatUpToDecimals(value, maxDecimals)\r\n
else -> formatToExactDecimals(value, precision(abs(value)))\r\n   } + unit.shortName()\r\n   }\r\n
}\r\n\r\n   private fun precision(value: Double): Int = when {\r\n   value < 1 -> 3\r\n   value < 10 -> 2\r\n
value < 100 -> 1\r\n   else -> 0\r\n   }\r\n\r\n  /**\r\n   * Returns a string representation of this duration value
expressed in the given [unit]\r\n   * and formatted with the specified [decimals] number of digits after decimal
point.\r\n   *\r\n   * Special cases:\r\n   * - the infinite duration is formatted as `"Infinity"` without unit\r\n
*/\r\n   * @return the value of duration in the specified [unit] followed by that unit abbreviated name: `d`, `h`, `m`,
`s`, `ms`, `us`, or `ns`.\r\n   *\r\n   * @throws IllegalArgumentException if [decimals] is less than zero.\r\n   */\r\n
* @sample samples.time.Durations.toStringDecimals\r\n   */\r\n   public fun toString(unit: DurationUnit,

```

```

decimals: Int = 0): String {
    require(decimals >= 0) { "decimals must be not negative, but was $decimals" }
    if (isInfinite()) return value.toString()
    val number = toDouble(unit)
    return when {
        abs(number) < 1e14 -> formatToExactDecimals(number, decimals.coerceAtMost(12))
        else -> formatScientific(number)
    } + unit.shortName()
}

/** Returns an ISO-8601 based string representation of this duration.
 * The returned value is presented in the format `PThHmMs.fS`, where `h`, `m`, `s` are the integer components of this duration (see [toComponents]) and `f` is a fractional part of second. Depending on the roundness of the value the fractional part can be formatted with either 0, 3, 6, or 9 decimal digits.
 * If the hours component absolute value of this duration is greater than [Int.MAX_VALUE], it is replaced with [Int.MAX_VALUE] so the infinite duration is formatted as `PT2147483647H`.
 * Negative durations are indicated with the sign `-` in the beginning of the returned string, for example, `-PT5M30S`.
 * @sample samples.time.Durations.toIsoString
 */
public fun toIsoString(): String = buildString {
    if (isNegative()) append('-')
    append("PT")
    absoluteValue.toComponents { hours, minutes, seconds, nanoseconds ->
        val hasHours = hours != 0
        val hasSeconds = seconds != 0 || nanoseconds != 0
        val hasMinutes = minutes != 0 || (hasSeconds && hasHours)
        if (hasHours) {
            append(hours).append('H')
        }
        if (hasMinutes) {
            append(minutes).append('M')
        }
        if (hasSeconds || (hasHours && !hasMinutes)) {
            append(seconds)
            if (nanoseconds != 0) {
                append('.')
                val nss = nanoseconds.toString().padStart(9, '0')
                when {
                    nanoseconds % 1_000_000 == 0 -> append(nss, 0, 3)
                    nanoseconds % 1_000 == 0 -> append(nss, 0, 6)
                    else -> append(nss)
                }
                append('S')
            }
        }
    }
}

// constructing from number of units
// extension functions
/** Returns a [Duration] equal to this [Int] number of the specified [unit].
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public fun Int.toDuration(unit: DurationUnit): Duration = toDouble().toDuration(unit)

/** Returns a [Duration] equal to this [Long] number of the specified [unit].
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public fun Long.toDuration(unit: DurationUnit): Duration = toDouble().toDuration(unit)

/** Returns a [Duration] equal to this [Double] number of the specified [unit].
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public fun Double.toDuration(unit: DurationUnit): Duration = Duration(convertDurationUnit(this, unit, storageUnit))

// constructing from number of units
// extension properties
/** Returns a [Duration] equal to this [Int] number of nanoseconds.
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public val Int.nanoseconds get() = toDuration(DurationUnit.NANOSECONDS)

/** Returns a [Duration] equal to this [Long] number of nanoseconds.
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public val Long.nanoseconds get() = toDuration(DurationUnit.NANOSECONDS)

/** Returns a [Duration] equal to this [Double] number of nanoseconds.
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public val Double.nanoseconds get() = toDuration(DurationUnit.NANOSECONDS)

/** Returns a [Duration] equal to this [Int] number of microseconds.
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public val Int.microseconds get() = toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Long] number of microseconds.
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public val Long.microseconds get() = toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Double] number of microseconds.
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public val Double.microseconds get() = toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Int] number of milliseconds.
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public val Int.milliseconds get() = toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Long] number of milliseconds.
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public val Long.milliseconds get() = toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Double] number of milliseconds.
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public val Double.milliseconds get() = toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Int] number of seconds.
 * @since Kotlin("1.3")
 * @ExperimentalTime
 */
public val Int.seconds get() =

```



```

when (this) {
    DurationUnit.NANOSECONDS -> "ns"
    DurationUnit.MICROSECONDS -> "us"
    DurationUnit.MILLISECONDS -> "ms"
    DurationUnit.SECONDS -> "s"
    DurationUnit.MINUTES -> "m"
    DurationUnit.HOURS -> "h"
    DurationUnit.DAYS -> "d"
}
"/**
 * Copyright 2010-2019
 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the
 Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package kotlin.time
import kotlin.annotation.AnnotationTarget

/**
 * This annotation marks the experimental preview of the
 standard library API for measuring time and working with durations.
 *
 * > Note that this API is in a preview
 state and has a very high chance of being changed in the future.
 * Do not use it if you develop a library since
 your library will become binary incompatible
 * with the future versions of the standard library.
 *
 * Any
 usage of a declaration annotated with `@ExperimentalTime` must be accepted either by
 * annotating that usage
 with the [UseExperimental] annotation, e.g. `@UseExperimental(ExperimentalTime::class)`,
 * or by using the
 compiler argument `Xuse-experimental=kotlin.time.ExperimentalTime`.
 */
@Experimental(level =
    Experimental.Level.ERROR)
@Retention(AnnotationRetention.BINARY)
@Target(
    CLASS,
    ANNOTATION_CLASS,
    PROPERTY,
    FIELD,
    LOCAL_VARIABLE,
    VALUE_PARAMETER,
    CONSTRUCTOR,
    FUNCTION,
    PROPERTY_GETTER,
    PROPERTY_SETTER,
    TYPEALIAS)
@SinceKotlin("1.3")
public annotation class
ExperimentalTime

"/**
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
 contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 license/LICENSE.txt file.
 */
package kotlin.time
import kotlin.contracts.*

/**
 * Executes
 the given function [block] and returns the duration of elapsed time interval.
 *
 * The elapsed time is measured
 with [MonoClock].
 */
@SinceKotlin("1.3")
@ExperimentalTime
public inline fun
measureTime(block: () -> Unit): Duration {
    contract {
        callsInPlace(block,
    InvocationKind.EXACTLY_ONCE)
    }
    return MonoClock.measureTime(block)
}

"/**
 * Executes the given function [block] and returns the duration of elapsed time interval.
 *
 * The elapsed time is
 measured with the specified `this` [Clock] instance.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public inline fun Clock.measureTime(block: () -> Unit):
Duration {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    val
    mark = markNow()
    block()
    return mark.elapsedNow()
}

/**
 * Data class representing a
 result of executing an action, along with the duration of elapsed time interval.
 *
 * @property value the result
 of the action.
 * @property duration the time elapsed to execute the action.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public data class TimedValue<T>(val value: T, val duration:
Duration)

/**
 * Executes the given function [block] and returns an instance of [TimedValue] class,
 containing both
 * the result of the function execution and the duration of elapsed time interval.
 *
 * The
 elapsed time is measured with [MonoClock].
 */
@SinceKotlin("1.3")
@ExperimentalTime
public
inline fun <T> measureTimedValue(block: () -> T): TimedValue<T> {
    contract {
        callsInPlace(block,
    InvocationKind.EXACTLY_ONCE)
    }
    return
    MonoClock.measureTimedValue(block)
}

/**
 * Executes the given [block] and returns an instance of
 [TimedValue] class, containing both
 * the result of function execution and the duration of elapsed time
 interval.
 *
 * The elapsed time is measured with the specified `this` [Clock] instance.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public inline fun <T> Clock.measureTimedValue(block: () -
> T): TimedValue<T> {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    val
    mark = markNow()
    val result = block()
    return TimedValue(result,
    mark.elapsedNow())
}

"/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 license/LICENSE.txt file.
 */
package kotlin
import kotlin.internal.InlineOnly

/**
 * Returns a hash code value for the object or zero if the object is `null`.
 *
 * @see Any.hashCode
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun Any?.hashCode(): Int = this?.hashCode() ?:
0
"/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of

```

```

this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*/\r\n\r\npackage kotlin\r\n\r\n/**\r\n * Represents a version of the Kotlin standard library.\r\n * [major], [minor] and [patch] are integer components of a version,\r\n * they must be non-negative and not greater than 255 ([MAX_COMPONENT_VALUE]).\r\n * @constructor Creates a version from all three components.\r\n */\r\n@SinceKotlin("1.1")\r\npublic class KotlinVersion(val major: Int, val minor: Int, val patch: Int) :
Comparable<KotlinVersion> {\r\n    /**\r\n     * Creates a version from [major] and [minor] components, leaving [patch] component zero.\r\n     */\r\n    public constructor(major: Int, minor: Int) : this(major, minor, 0)\r\n\r\n    private val version = versionOf(major, minor, patch)\r\n\r\n    private fun versionOf(major: Int, minor: Int, patch: Int): Int {\r\n        require(major in 0..MAX_COMPONENT_VALUE && minor in 0..MAX_COMPONENT_VALUE && patch in 0..MAX_COMPONENT_VALUE) {\r\n            "Version components are out of range: $major.$minor.$patch"\r\n        }\r\n        return major.shl(16) + minor.shl(8) + patch\r\n    }\r\n\r\n    /**\r\n     * Returns the string representation of this version\r\n     */\r\n    override fun toString(): String = "$major.$minor.$patch"\r\n\r\n    override fun equals(other: Any?): Boolean {\r\n        if (this === other) return true\r\n        val otherVersion = (other as? KotlinVersion) ?: return false\r\n        return this.version == otherVersion.version\r\n    }\r\n\r\n    override fun hashCode(): Int = version\r\n\r\n    override fun compareTo(other: KotlinVersion): Int = version - other.version\r\n\r\n    /**\r\n     * Returns `true` if this version is not less than the version specified\r\n     * with the provided [major] and [minor] components.\r\n     */\r\n    public fun isAtLeast(major: Int, minor: Int): Boolean = // this.version >= versionOf(major, minor, 0)\r\n        this.major > major || (this.major == major && this.minor >= minor)\r\n\r\n    /**\r\n     * Returns `true` if this version is not less than the version specified\r\n     * with the provided [major], [minor] and [patch] components.\r\n     */\r\n    public fun isAtLeast(major: Int, minor: Int, patch: Int): Boolean = // this.version >= versionOf(major, minor, patch)\r\n        this.major > major || (this.major == major && (this.minor > minor || this.minor == minor && this.patch >= patch))\r\n\r\n    companion object {\r\n        /**\r\n         * Maximum value a version component can have, a constant value 255.\r\n         */\r\n        // NOTE: Must be placed before CURRENT because its initialization requires this field being initialized in JS\r\n        public const val MAX_COMPONENT_VALUE = 255\r\n\r\n        /**\r\n         * Returns the current version of the Kotlin standard library.\r\n         */\r\n        @kotlin.jvm.JvmField\r\n        public val CURRENT: KotlinVersion = KotlinVersion(1, 3, 50) // value is written here automatically during build\r\n    }\r\n\r\n    "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n@file:kotlin.jvm.JvmName("LateinitKt")\r\n@file:Suppress("unused")\r\n\r\npackage kotlin\r\n\r\n\r\nimport kotlin.internal.InlineOnly\r\nimport kotlin.internal.AccessibleLateinitPropertyLiteral\r\nimport kotlin.reflect.KProperty0\r\n\r\n\r\n/**\r\n * Returns `true` if this lateinit property has been assigned a value, and `false` otherwise.\r\n * Cannot be used in an inline function, to avoid binary compatibility issues.\r\n */\r\n\r\n@SinceKotlin("1.2")\r\n@InlineOnly\r\ninline val @receiver:AccessibleLateinitPropertyLiteral KProperty0<*>.isInitialized: Boolean\r\n    get() = throw NotImplementedError("Implementation is intrinsic")\r\n\r\n"/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n */\r\n\r\n@file:kotlin.jvm.JvmName("LazyKt")\r\n@file:kotlin.jvm.JvmMultifileClass\r\n\r\npackage kotlin\r\n\r\n\r\nimport kotlin.reflect.KProperty\r\n\r\n\r\n/**\r\n * Represents a value with lazy initialization.\r\n * To create an instance of [Lazy] use the [lazy] function.\r\n * public interface Lazy<out T> {\r\n *     /**\r\n *      * Gets the lazily initialized value of the current Lazy instance.\r\n *      * Once the value was initialized it must not change during the rest of lifetime of this Lazy instance.\r\n *      */\r\n *     public val value: T\r\n *     /**\r\n *      * Returns `true` if a value for this Lazy instance has been already initialized, and `false` otherwise.\r\n *      * Once this function has returned `true` it stays `true` for the rest of lifetime of this Lazy instance.\r\n *      */\r\n *     public fun isInitialized(): Boolean\r\n * }\r\n\r\n\r\n/**\r\n * Creates a new instance of the [Lazy] that is already initialized with the specified [value].\r\n */\r\n\r\npublic fun <T> lazyOf(value: T): Lazy<T> = InitializedLazyImpl(value)\r\n\r\n\r\n/**\r\n * An extension to delegate a read-only property of type [T] to an instance of [Lazy].\r\n * This extension allows to

```

```

use instances of Lazy for property delegation:\r\n * `val property: String by lazy { initializer }`\r\n
*^/\r\n@kotlin.internal.InlineOnly\r\npublic inline operator fun <T> Lazy<T>.getValue(thisRef: Any?, property:
KProperty<*>): T = value\r\n\r\n/**\r\n * Specifies how a [Lazy] instance synchronizes initialization among
multiple threads.\r\n *^/\r\npublic enum class LazyThreadSafetyMode {\r\n\r\n    /**\r\n     * Locks are used to ensure
that only a single thread can initialize the [Lazy] instance.\r\n     *^/\r\n    SYNCHRONIZED,\r\n\r\n    /**\r\n     *
Initializer function can be called several times on concurrent access to uninitialized [Lazy] instance value,\r\n     *
but only the first returned value will be used as the value of [Lazy] instance.\r\n     *^/\r\n    PUBLICATION,\r\n\r\n\r\n    /**\r\n     * No locks are used to synchronize an access to the [Lazy] instance value; if the instance is accessed from
multiple threads, its behavior is undefined.\r\n     *^/\r\n     * This mode should not be used unless the [Lazy] instance
is guaranteed never to be initialized from more than one thread.\r\n     *^/\r\n    NONE,\r\n\r\n}\r\n\r\n\r\ninternal object
UNINITIALIZED_VALUE\r\n\r\n// internal to be called from lazy in JS\r\ninternal class UnsafeLazyImpl<out
T>(initializer: () -> T) : Lazy<T>, Serializable {\r\n    private var initializer: (() -> T)? = initializer\r\n    private var
_value: Any? = UNINITIALIZED_VALUE\r\n\r\n    override val value: T\r\n        get() {\r\n            if (_value ===
UNINITIALIZED_VALUE) {\r\n                _value = initializer!!()\r\n                initializer = null\r\n            }\r\n
            @Suppress("\r\n                UNCHECKED_CAST\r\n            ") return _value as T\r\n        }\r\n\r\n    override fun isInitialized():
Boolean = _value != UNINITIALIZED_VALUE\r\n\r\n    override fun toString(): String = if (isInitialized())
value.toString() else "\r\n                Lazy value not initialized yet.\r\n            "\r\n\r\n    private fun writeReplace(): Any =
InitializedLazyImpl(value)\r\n}\r\n\r\ninternal class InitializedLazyImpl<out T>(override val value: T) : Lazy<T>,
Serializable {\r\n\r\n    override fun isInitialized(): Boolean = true\r\n\r\n    override fun toString(): String =
value.toString()\r\n\r\n}\r\n\r\n"/**\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n
*^/\r\n\r\n@file:kotlin.jvm.JvmMultifileClass\r\n@file:kotlin.jvm.JvmName("NumbersKt")\r\npackage
kotlin\r\n\r\n/**\r\n * Counts the number of set bits in the binary representation of this [Int] number.\r\n
*^/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic expect fun Int.countOneBits(): Int\r\n\r\n/**\r\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int]
number.\r\n *^/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic expect fun
Int.countLeadingZeroBits(): Int\r\n\r\n/**\r\n * Counts the number of consecutive least significant bits that are zero
in the binary representation of this [Int] number.\r\n
*^/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic expect fun Int.countTrailingZeroBits():
Int\r\n\r\n/**\r\n * Returns a number having a single bit set in the position of the most significant set bit of this [Int]
number,\r\n * or zero, if this number is zero.\r\n *^/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic
expect fun Int.takeHighestOneBit(): Int\r\n\r\n/**\r\n * Returns a number having a single bit set in the position of
the least significant set bit of this [Int] number,\r\n * or zero, if this number is zero.\r\n
*^/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic expect fun Int.takeLowestOneBit():
Int\r\n\r\n/**\r\n * Rotates the binary representation of this [Int] number left by the specified [bitCount] number of
bits.\r\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on
the right side.\r\n * Rotating the number left by a negative bit count is the same as rotating it right by the
negated bit count:\r\n * `number.rotateLeft(-n) == number.rotateRight(n)`\r\n * Rotating by a multiple of
[Int.SIZE_BITS] (32) returns the same number, or more generally\r\n * `number.rotateLeft(n) ==
number.rotateLeft(n % 32)`\r\n *^/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic expect fun
Int.rotateLeft(bitCount: Int): Int\r\n\r\n/**\r\n * Rotates the binary representation of this [Int] number right by the
specified [bitCount] number of bits.\r\n * The least significant bits pushed out from the right side reenter the number
as the most significant bits on the left side.\r\n * Rotating the number right by a negative bit count is the same
as rotating it left by the negated bit count:\r\n * `number.rotateRight(-n) == number.rotateLeft(n)`\r\n * Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally\r\n *
`number.rotateRight(n) == number.rotateRight(n % 32)`\r\n
*^/\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\npublic expect fun Int.rotateRight(bitCount: Int):
Int

```


returns the same number, or more generally

```
number.rotateLeft(n) == number.rotateLeft(n % 8)
```

`@SinceKotlin("1.3")@ExperimentalStdlibApi`

```
public fun Byte.rotateLeft(bitCount: Int): Byte =
    (toInt().shl(bitCount and 7) or (toInt() and 0xFF).ushr(8 - (bitCount and 7))).toByte()

```

 Rotates the binary representation of this [Byte] number right by the specified [bitCount] number of bits. The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side. Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.


```
number.rotateRight(-n) == number.rotateLeft(n)
```

 Rotating by a multiple of [Byte.SIZE_BITS] (8) returns the same number, or more generally


```
number.rotateRight(n) == number.rotateRight(n % 8)
```

`@SinceKotlin("1.3")@ExperimentalStdlibApi`

```
public fun Byte.rotateRight(bitCount: Int): Byte =
    (toInt().shl(8 - (bitCount and 7)) or (toInt() and 0xFF).ushr(bitCount and 7)).toByte()

```

 Counts the number of set bits in the binary representation of this [Short] number.
 `@SinceKotlin("1.3")@ExperimentalStdlibApi`

```
internal inline fun
Short.countOneBits(): Int = (toInt() and 0xFFFF).countOneBits()

```

 Counts the number of consecutive most significant bits that are zero in the binary representation of this [Short] number.
 `@SinceKotlin("1.3")@ExperimentalStdlibApi`

```
internal inline fun
Short.countLeadingZeroBits(): Int =
    (toInt() and 0xFFFF).countLeadingZeroBits() - (Int.SIZE_BITS -
    Short.SIZE_BITS)

```

 Counts the number of consecutive least significant bits that are zero in the binary representation of this [Short] number.
 `@SinceKotlin("1.3")@ExperimentalStdlibApi`

```
internal inline fun
Short.countTrailingZeroBits(): Int = (toInt() or 0x10000).countTrailingZeroBits()

```

 Returns a number having a single bit set in the position of the most significant set bit of this [Short] number, or zero, if this number is zero.
 `@SinceKotlin("1.3")@ExperimentalStdlibApi`

```
internal inline fun
Short.takeHighestOneBit(): Short = (toInt() and 0xFFFF).takeHighestOneBit().toShort()

```

 Returns a number having a single bit set in the position of the least significant set bit of this [Short] number, or zero, if this number is zero.
 `@SinceKotlin("1.3")@ExperimentalStdlibApi`

```
internal inline fun
Short.takeLowestOneBit(): Short = toInt().takeLowestOneBit().toShort()

```

 Rotates the binary representation of this [Short] number left by the specified [bitCount] number of bits. The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side. Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count.


```
number.rotateLeft(-n) == number.rotateRight(n)
```

 Rotating by a multiple of [Short.SIZE_BITS] (16) returns the same number, or more generally


```
number.rotateLeft(n) == number.rotateLeft(n % 16)
```

`@SinceKotlin("1.3")@ExperimentalStdlibApi`

```
public fun Short.rotateLeft(bitCount: Int): Short =
    (toInt().shl(bitCount and 15) or (toInt() and 0xFFFF).ushr(16 - (bitCount and 15))).toShort()

```

 Rotates the binary representation of this [Short] number right by the specified [bitCount] number of bits. The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side. Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.


```
number.rotateRight(-n) == number.rotateLeft(n)
```

 Rotating by a multiple of [Short.SIZE_BITS] (16) returns the same number, or more generally


```
number.rotateRight(n) == number.rotateRight(n % 16)
```

`@SinceKotlin("1.3")@ExperimentalStdlibApi`

```
public fun Short.rotateRight(bitCount: Int): Short =
    (toInt().shl(16 - (bitCount and 15)) or (toInt() and 0xFFFF).ushr(bitCount and 15)).toShort()

```

 Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 `package`

```
kotlin
import kotlin.internal.RequireKotlin
import
kotlin.internal.RequireKotlinVersionKind
import kotlin.internal.InlineOnly
@SinceKotlin("1.2")@Suppress
("INVISIBLE_MEMBER", "INVISIBLE_REFERENCE")
@RequireKotlin("1.2.30", level =
DeprecationLevel.HIDDEN, versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
public inline

```

```

fun <R> suspend(noinline block: suspend () -> R): suspend () -> R = block\r\n", "/*\r\n * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.\r\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\r\n
*/\r\n\r\n@file:kotlin.jvm.JvmName("\TuplesKt")\r\n\r\npackage kotlin\r\n\r\n\r\n/*\r\n * Represents a generic
pair of two values.\r\n * There is no meaning attached to values in this class, it can be used for any purpose.\r\n
* Pair exhibits value semantics, i.e. two pairs are equal if both components are equal.\r\n * An example of
decomposing it into values:\r\n * @sample samples.misc.Tuples.pairDestructuring\r\n * @param A type of the
first value.\r\n * @param B type of the second value.\r\n * @property first First value.\r\n * @property second
Second value.\r\n * @constructor Creates a new instance of Pair.\r\n */\r\n\r\npublic data class Pair<out A, out B>(\r\n
public val first: A,\r\n public val second: B\r\n) : Serializable {\r\n\r\n /*\r\n * Returns string representation
of the [Pair] including its [first] and [second] values.\r\n * \r\n public override fun toString(): String = "$first,
$second"\r\n}\r\n\r\n/*\r\n * Creates a tuple of type [Pair] from this and [that].\r\n * This can be useful for
creating [Map] literals with less noise, for example:\r\n * @sample
samples.collections.Maps.Instantiation.mapFromPairs\r\n */\r\n\r\npublic infix fun <A, B> A.to(that: B): Pair<A, B> =
Pair(this, that)\r\n\r\n/*\r\n * Converts this pair into a list.\r\n * @sample samples.misc.Tuples.pairToList\r\n
*/\r\n\r\npublic fun <T> Pair<T, T>.toList(): List<T> = listOf(first, second)\r\n\r\n/*\r\n * Represents a triad of
values\r\n * There is no meaning attached to values in this class, it can be used for any purpose.\r\n * Triple
exhibits value semantics, i.e. two triples are equal if all three components are equal.\r\n * An example of
decomposing it into values:\r\n * @sample samples.misc.Tuples.tripleDestructuring\r\n * @param A type of
the first value.\r\n * @param B type of the second value.\r\n * @param C type of the third value.\r\n * @property
first First value.\r\n * @property second Second value.\r\n * @property third Third value.\r\n */\r\n\r\npublic data class
Triple<out A, out B, out C>(\r\n public val first: A,\r\n public val second: B,\r\n public val third: C\r\n) :
Serializable {\r\n\r\n /*\r\n * Returns string representation of the [Triple] including its [first], [second] and
[third] values.\r\n * \r\n public override fun toString(): String = "$first, $second, $third"\r\n}\r\n\r\n/*\r\n
Converts this triple into a list.\r\n * @sample samples.misc.Tuples.tripleToList\r\n */\r\n\r\npublic fun <T> Triple<T, T,
T>.toList(): List<T> = listOf(first, second, third)\r\n", "/*\r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n */\r\n\r\n@file:Suppress("INVISIBLE_REFERENCE",
"INVISIBLE_MEMBER")\r\n\r\npackage kotlin\r\n\r\nimport kotlin.annotation.AnnotationTarget.*\r\nimport
kotlin.internal.RequireKotlin\r\nimport kotlin.internal.RequireKotlinVersionKind\r\n\r\n/*\r\n * Marks the API that
is dependent on the experimental unsigned types, including those types themselves.\r\n * Usages of such API
will be reported as warnings unless an explicit opt-in with\r\n * the [UseExperimental] annotation, e.g.
`@UseExperimental(ExperimentalUnsignedTypes::class)`,\r\n * or with the `-Xuse-
experimental=kotlin.ExperimentalUnsignedTypes` compiler option is given.\r\n * It's recommended to
propagate the experimental status to the API that depends on unsigned types by annotating it with this
annotation.\r\n * \r\n @Experimental(level =
Experimental.Level.WARNING)\r\n @MustBeDocumented\r\n @Target(CLASS, ANNOTATION_CLASS,
PROPERTY, FIELD, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION,
PROPERTY_GETTER, PROPERTY_SETTER,
TYPEALIAS)\r\n @Retention(AnnotationRetention.BINARY)\r\n @RequireKotlin("1.2.50", versionKind =
RequireKotlinVersionKind.COMPILER_VERSION)\r\n\r\npublic annotation class
ExperimentalUnsignedTypes\r\n", "/*\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\r\n */\r\n\r\n// Auto-generated file. DO NOT EDIT!\r\n\r\npackage
kotlin.ranges\r\n\r\n\r\nimport kotlin.internal.*\r\n\r\n/*\r\n * A range of values of type `UInt`.\r\n
*/\r\n\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic class UIntRange(start: UInt, endInclusive:
UInt) : UIntProgression(start, endInclusive, 1), ClosedRange<UInt> {\r\n public override val start: UInt get() = first\r\n
public override val endInclusive: UInt get() = last\r\n\r\n public override fun contains(value: UInt): Boolean = first <= value &&

```

```

value <= last\r\n\r\n override fun isEmpty(): Boolean = first > last\r\n\r\n override fun equals(other: Any?):
Boolean =\r\n other is UIntRange && (isEmpty() && other.isEmpty()) ||\r\n first == other.first && last
== other.last)\r\n\r\n override fun hashCode(): Int =\r\n if (isEmpty()) -1 else (31 * first.toInt() +
last.toInt())\r\n\r\n override fun toString(): String = \"$first..$last\"\r\n\r\n companion object {\r\n /** An
empty range of values of type UInt. */\r\n public val EMPTY: UIntRange = UIntRange(UInt.MAX_VALUE,
UInt.MIN_VALUE)\r\n }\r\n}\r\n\r\n/** A progression of values of type `UInt`.
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic open class UIntProgression\r\ninternal
constructor(\r\n start: UInt,\r\n endInclusive: UInt,\r\n step: Int\r\n) : Iterable<UInt> {\r\n init {\r\n if
(step == 0.toInt()) throw kotlin.IllegalArgumentException("Step must be non-zero.")\r\n if (step ==
Int.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Int.MIN_VALUE to avoid
overflow on negation.")\r\n }\r\n\r\n /**\r\n * The first element in the progression.\r\n */\r\n public val
first: UInt = start\r\n\r\n /**\r\n * The last element in the progression.\r\n */\r\n public val last: UInt =
getProgressionLastElement(start, endInclusive, step)\r\n\r\n /**\r\n * The step of the progression.\r\n */\r\n
public val step: Int = step\r\n\r\n override fun iterator(): UIntIterator = UIntProgressionIterator(first, last,
step)\r\n\r\n /** Checks if the progression is empty. */\r\n public open fun isEmpty(): Boolean = if (step > 0) first
> last else first < last\r\n\r\n override fun equals(other: Any?): Boolean =\r\n other is UIntProgression &&
(isEmpty() && other.isEmpty()) ||\r\n first == other.first && last == other.last && step ==
other.step)\r\n\r\n override fun hashCode(): Int =\r\n if (isEmpty()) -1 else (31 * (31 * first.toInt() +
last.toInt()) + step.toInt())\r\n\r\n override fun toString(): String = if (step > 0) \"$first..$last step $step\" else
\"$first downTo $last step ${-step}\"\r\n\r\n companion object {\r\n /**\r\n * Creates UIntProgression
within the specified bounds of a closed range.\r\n\r\n * The progression starts with the [rangeStart] value and
goes toward the [rangeEnd] value not excluding it, with the specified [step].\r\n\r\n * In order to go backwards the
[step] must be negative.\r\n\r\n * [step] must be greater than `Int.MIN_VALUE` and not equal to
zero.\r\n\r\n */\r\n public fun fromClosedRange(rangeStart: UInt, rangeEnd: UInt, step: Int): UIntProgression
= UIntProgression(rangeStart, rangeEnd, step)\r\n }\r\n}\r\n\r\n/** An iterator over a progression of
values of type `UInt`.
*/\r\n * @property step the number by which the value is incremented on each step.\r\n
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\nprivate class UIntProgressionIterator(first: UInt,
last: UInt, step: Int) : UIntIterator() {\r\n private val finalElement = last\r\n private var hasNext: Boolean = if
(step > 0) first <= last else first >= last\r\n private val step = step.toInt() // use 2-complement math for negative
steps\r\n private var next = if (hasNext) first else finalElement\r\n\r\n override fun hasNext(): Boolean =
hasNext\r\n\r\n override fun nextUInt(): UInt {\r\n val value = next\r\n if (value == finalElement) {\r\n
if (!hasNext) throw kotlin.NoSuchElementException()\r\n hasNext = false\r\n } else {\r\n next
+= step\r\n }\r\n return value\r\n }\r\n}\r\n\r\n"/**\r\n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin
Programming Language contributors.\r\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\r\n */\r\n\r\n// Auto-generated file. DO NOT EDIT!\r\n\r\npackage
kotlin.collections\r\n\r\n/** An iterator over a sequence of values of type `UByte`.
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic abstract class UByteIterator :
Iterator<UByte> {\r\n override final fun next() = nextUByte()\r\n\r\n /** Returns the next value in the sequence
without boxing. */\r\n public abstract fun nextUByte(): UByte\r\n}\r\n\r\n/** An iterator over a sequence of values
of type `UShort`.
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic abstract class
UShortIterator : Iterator<UShort> {\r\n override final fun next() = nextUShort()\r\n\r\n /** Returns the next
value in the sequence without boxing. */\r\n public abstract fun nextUShort(): UShort\r\n}\r\n\r\n/** An iterator
over a sequence of values of type `UInt`.
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic
abstract class UIntIterator : Iterator<UInt> {\r\n override final fun next() = nextUInt()\r\n\r\n /** Returns the
next value in the sequence without boxing. */\r\n public abstract fun nextUInt(): UInt\r\n}\r\n\r\n/** An iterator
over a sequence of values of type `ULong`.
*/\r\n@SinceKotlin("1.3")\r\n@ExperimentalUnsignedTypes\r\npublic
abstract class ULongIterator : Iterator<ULong> {\r\n override final fun next() = nextULong()\r\n\r\n /** Returns
the next value in the sequence without boxing. */\r\n public abstract fun nextULong(): ULong\r\n}\r\n\r\n"/**

```

```

* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
* Auto-generated file. DO NOT EDIT!
package kotlin.ranges
import kotlin.internal.*

A range of values of type `ULong`.
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public class ULongRange(start: ULong, endInclusive: ULong) : ULongProgression(start, endInclusive, 1),
    ClosedRange<ULong> {
    override val start: ULong get() = first
    override val endInclusive: ULong get() = last
    override fun contains(value: ULong): Boolean = first <= value && value <= last
    override fun isEmpty(): Boolean = first > last
    override fun equals(other: Any?): Boolean =
        other is ULongRange && (isEmpty() && other.isEmpty() ||
            first == other.first && last == other.last)
    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt())
    override fun toString(): String = "$first..$last"
    companion object {
        /** An empty range of values of type ULong. */
        public val EMPTY: ULongRange =
            ULongRange(ULong.MAX_VALUE, ULong.MIN_VALUE)
    }
}

A progression of values of type `ULong`.
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public open class ULongProgression(
    internal constructor(
        start: ULong,
        endInclusive: ULong,
        step: Long
    ) :
    Iterable<ULong> {
    init {
        if (step == 0.toLong()) throw kotlin.IllegalArgumentException("Step must be non-zero.")
        if (step == Long.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Long.MIN_VALUE to avoid overflow on negation.")
    }
    /** The first element in the progression. */
    public val first: ULong = start
    /** The last element in the progression. */
    public val last: ULong = getProgressionLastElement(start, endInclusive, step)
    /** The step of the progression. */
    public val step: Long = step
    override fun iterator(): ULongIterator = ULongProgressionIterator(first, last, step)
    /** Checks if the progression is empty. */
    public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last
    override fun equals(other: Any?): Boolean =
        other is ULongProgression && (isEmpty() && other.isEmpty() ||
            first == other.first && last == other.last && step == other.step)
    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt()) + (step xor (step ushr 32)).toInt())
    override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step ${-step}"
    companion object {
        /** Creates ULongProgression within the specified bounds of a closed range.
            * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].
            * In order to go backwards the [step] must be negative.
            * [step] must be greater than `Long.MIN_VALUE` and not equal to zero.
            */
        public fun fromClosedRange(rangeStart: ULong, rangeEnd: ULong, step: Long): ULongProgression =
            ULongProgression(rangeStart, rangeEnd, step)
    }
}

An iterator over a progression of values of type `ULong`.
@property step the number by which the value is incremented on each step.
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
private class ULongProgressionIterator(first: ULong, last: ULong, step: Long) : ULongIterator() {
    private val finalElement = last
    private var hasNext: Boolean = if (step > 0) first <= last else first >= last
    private val step = step.toULong() // use 2-complement math for negative steps
    private var next = if (hasNext) first else finalElement
    override fun hasNext(): Boolean = hasNext
    override fun nextULong(): ULong {
        val value = next
        if (value == finalElement) {
            if (!hasNext) throw kotlin.NoSuchElementException()
            hasNext = false
        } else {
            next += step
        }
        return value
    }
}

* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
* Returns the smaller of two values.
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun min(a: UInt, b: UInt): UInt {
    return minOf(a, b)
}

* Returns the smaller of two values.
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun min(a: ULong, b: ULong): ULong {
    return minOf(a, b)
}

* Returns the greater of two

```


Counts the number of consecutive least significant bits that are zero in the binary representation of this [ULong] number.

```
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ULong.countTrailingZeroBits(): Int = toLong().countTrailingZeroBits()\r\n\r\n**\r\n * Returns a number having a single bit set in the position of the most significant set bit of this [ULong] number,\r\n * or zero, if this number is zero.
```

```
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ULong.takeHighestOneBit(): ULong =\r\ntoLong().takeHighestOneBit().toULong()\r\n\r\n**\r\n * Returns a number having a single bit set in the position of\r\n the least significant set bit of this [ULong] number,\r\n * or zero, if this number is zero.
```

```
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ULong.takeLowestOneBit(): ULong =\r\ntoLong().takeLowestOneBit().toULong()\r\n\r\n**\r\n * Rotates the binary representation of this [ULong] number\r\n left by the specified [bitCount] number of bits.\r\n * The most significant bits pushed out from the left side reenter\r\n the number as the least significant bits on the right side.\r\n * Rotating the number left by a negative bit count\r\n is the same as rotating it right by the negated bit count:\r\n * `number.rotateLeft(-n) == number.rotateRight(n)`\r\n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally\r\n * `number.rotateLeft(n) == number.rotateLeft(n % 64)`
```

```
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ULong.rotateLeft(bitCount: Int): ULong =\r\ntoLong().rotateLeft(bitCount).toULong()\r\n\r\n**\r\n * Rotates the binary representation of this [ULong] number\r\n right by the specified [bitCount] number of bits.\r\n * The least significant bits pushed out from the right side reenter\r\n the number as the most significant bits on the left side.\r\n * Rotating the number right by a negative bit count\r\n is the same as rotating it left by the negated bit count:\r\n * `number.rotateRight(-n) == number.rotateLeft(n)`\r\n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally\r\n * `number.rotateRight(n) == number.rotateRight(n % 64)`
```

```
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun ULong.rotateRight(bitCount: Int): ULong =\r\ntoLong().rotateRight(bitCount).toULong()\r\n\r\n**\r\n * Counts the number of set bits in the binary representation\r\n of this [UByte] number.
```

```
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun UByte.countOneBits(): Int = toUInt().countOneBits()\r\n\r\n**\r\n * Counts the number\r\n of consecutive most significant bits that are zero in the binary representation of this [UByte] number.
```

```
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun UByte.countLeadingZeroBits(): Int = toByte().countLeadingZeroBits()\r\n\r\n**\r\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UByte]\r\n number.
```

```
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun UByte.countTrailingZeroBits(): Int = toByte().countTrailingZeroBits()\r\n\r\n**\r\n * Returns a number having a single bit set in the position of the most significant set bit of this [UByte] number,\r\n * or zero, if this number is zero.
```

```
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun UByte.takeHighestOneBit(): UByte =\r\ntoInt().takeHighestOneBit().toUByte()\r\n\r\n**\r\n * Returns a number having a single bit set in the position of the\r\n least significant set bit of this [UByte] number,\r\n * or zero, if this number is zero.
```

```
*\r\n@SinceKotlin("1.3")\r\n@ExperimentalStdlibApi\r\n@ExperimentalUnsignedTypes\r\n@kotlin.internal.InlineOnly\r\npublic inline fun UByte.takeLowestOneBit(): UByte =\r\ntoInt().takeLowestOneBit().toUByte()\r\n\r\n**\r\n * Rotates the binary representation of this [UByte] number
```

left by the specified [bitCount] number of bits.\r\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\r\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\r\n * `number.rotateLeft(-n) == number.rotateRight(n)`\r\n * Rotating by a multiple of [UByte.SIZE_BITS] (8) returns the same number, or more generally\r\n * `number.rotateLeft(n) == number.rotateLeft(n % 8)`\r\n *\r\n @SinceKotlin("1.3")\r\n @ExperimentalStdlibApi\r\n @ExperimentalUnsignedTypes\r\n @kotlin.internal.InlineOnly\r\n public inline fun UByte.rotateLeft(bitCount: Int): UByte = toByte().rotateLeft(bitCount).toUByte()\r\n\r\n **\r\n * Rotates the binary representation of this [UByte] number right by the specified [bitCount] number of bits.\r\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\r\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\r\n * `number.rotateRight(-n) == number.rotateLeft(n)`\r\n * Rotating by a multiple of [UByte.SIZE_BITS] (8) returns the same number, or more generally\r\n * `number.rotateRight(n) == number.rotateRight(n % 8)`\r\n *\r\n @SinceKotlin("1.3")\r\n @ExperimentalStdlibApi\r\n @ExperimentalUnsignedTypes\r\n @kotlin.internal.InlineOnly\r\n public inline fun UByte.rotateRight(bitCount: Int): UByte = toByte().rotateRight(bitCount).toUByte()\r\n\r\n\r\n **\r\n * Counts the number of set bits in the binary representation of this [UShort] number.\r\n *\r\n @SinceKotlin("1.3")\r\n @ExperimentalStdlibApi\r\n @ExperimentalUnsignedTypes\r\n @kotlin.internal.InlineOnly\r\n public inline fun UShort.countOneBits(): Int = toUInt().countOneBits()\r\n\r\n\r\n **\r\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UShort] number.\r\n *\r\n @SinceKotlin("1.3")\r\n @ExperimentalStdlibApi\r\n @ExperimentalUnsignedTypes\r\n @kotlin.internal.InlineOnly\r\n public inline fun UShort.countLeadingZeroBits(): Int = toShort().countLeadingZeroBits()\r\n\r\n\r\n **\r\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UShort] number.\r\n *\r\n @SinceKotlin("1.3")\r\n @ExperimentalStdlibApi\r\n @ExperimentalUnsignedTypes\r\n @kotlin.internal.InlineOnly\r\n public inline fun UShort.countTrailingZeroBits(): Int = toShort().countTrailingZeroBits()\r\n\r\n\r\n **\r\n * Returns a number having a single bit set in the position of the most significant set bit of this [UShort] number, or zero, if this number is zero.\r\n *\r\n @SinceKotlin("1.3")\r\n @ExperimentalStdlibApi\r\n @ExperimentalUnsignedTypes\r\n @kotlin.internal.InlineOnly\r\n public inline fun UShort.takeHighestOneBit(): UShort = toInt().takeHighestOneBit().toUShort()\r\n\r\n\r\n **\r\n * Returns a number having a single bit set in the position of the least significant set bit of this [UShort] number, or zero, if this number is zero.\r\n *\r\n @SinceKotlin("1.3")\r\n @ExperimentalStdlibApi\r\n @ExperimentalUnsignedTypes\r\n @kotlin.internal.InlineOnly\r\n public inline fun UShort.takeLowestOneBit(): UShort = toInt().takeLowestOneBit().toUShort()\r\n\r\n\r\n\r\n **\r\n * Rotates the binary representation of this [UShort] number left by the specified [bitCount] number of bits.\r\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\r\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\r\n * `number.rotateLeft(-n) == number.rotateRight(n)`\r\n * Rotating by a multiple of [UShort.SIZE_BITS] (16) returns the same number, or more generally\r\n * `number.rotateLeft(n) == number.rotateLeft(n % 16)`\r\n *\r\n @SinceKotlin("1.3")\r\n @ExperimentalStdlibApi\r\n @ExperimentalUnsignedTypes\r\n @kotlin.internal.InlineOnly\r\n public inline fun UShort.rotateLeft(bitCount: Int): UShort = toShort().rotateLeft(bitCount).toUShort()\r\n\r\n\r\n\r\n **\r\n * Rotates the binary representation of this [UShort] number right by the specified [bitCount] number of bits.\r\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\r\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\r\n * `number.rotateRight(-n) == number.rotateLeft(n)`\r\n * Rotating by a multiple of [UShort.SIZE_BITS] (16) returns the same number, or more generally\r\n * `number.rotateRight(n) == number.rotateRight(n % 16)`\r\n *


```

*^r\n@SinceKotlin("1.3")r\n@ExperimentalStdlibApi\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShort.rotateRight(bitCount: Int): UShort =
toShort().rotateRight(bitCount).toUShort()\n", "/*r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin
Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be
found in the license/LICENSE.txt file.\n
*^r\n\n@file:UseExperimental(ExperimentalUnsignedTypes::class)\npackage kotlin.internal\n\n// (a - b) mod
c\nprivate fun differenceModulo(a: UInt, b: UInt, c: UInt): UInt {r\n    val ac = a % c\n    val bc = b % c\n
return if (ac >= bc) ac - bc else ac - bc + c\n}\n\nprivate fun differenceModulo(a: ULong, b: ULong, c: ULong):
ULong {r\n    val ac = a % c\n    val bc = b % c\n    return if (ac >= bc) ac - bc else ac - bc + c\n}\n\n/**r\n
* Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is
in the range\n * from [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative\n *
[step].\n * No validation on passed parameters is performed. The given parameters should satisfy the
condition:\n *\n * - either `step > 0` and `start <= end`,\n * - or `step < 0` and `start >= end`.\n *\n * @param
start first element of the progression\n * @param end ending bound for the progression\n * @param step
increment, or difference of successive elements in the progression\n * @return the final element of the
progression\n * @suppress\n *^r\n@PublishedApi\n@SinceKotlin("1.3")\ninternal fun
getProgressionLastElement(start: UInt, end: UInt, step: Int): UInt = when {r\n    step > 0 -> if (start >= end) end
else end - differenceModulo(end, start, step.toUInt())\n    step < 0 -> if (start <= end) end else end +
differenceModulo(start, end, (-step).toUInt())\n    else -> throw kotlin.IllegalArgumentException("Step is
zero.")\n}\n\n/**r\n * Calculates the final element of a bounded arithmetic progression, i.e. the last element of
the progression which is in the range\n * from [start] to [end] in case of a positive [step], or from [end] to [start] in
case of a negative\n * [step].\n * No validation on passed parameters is performed. The given parameters
should satisfy the condition:\n *\n * - either `step > 0` and `start <= end`,\n * - or `step < 0` and `start >=
end`.\n *\n * @param start first element of the progression\n * @param end ending bound for the
progression\n * @param step increment, or difference of successive elements in the progression\n * @return the
final element of the progression\n * @suppress\n *^r\n@PublishedApi\n@SinceKotlin("1.3")\ninternal fun
getProgressionLastElement(start: ULong, end: ULong, step: Long): ULong = when {r\n    step > 0 -> if (start >=
end) end else end - differenceModulo(end, start, step.toULong())\n    step < 0 -> if (start <= end) end else end +
differenceModulo(start, end, (-step).toULong())\n    else -> throw kotlin.IllegalArgumentException("Step is
zero.")\n}\n\n", "/*r\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt
file.\n
*^r\n\n@file:kotlin.jvm.JvmName("UStringsKt") // string representation of unsigned
numbers\n\npackage kotlin.text\n\n/**r\n * Returns a string representation of this [Byte] value in the
specified [radix].\n * @throws IllegalArgumentException when [radix] is not a valid radix for number to
string conversion.\n
*^r\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n//@kotlin.internal.InlineOnly\npublic /*inline*/
fun UByte.toString(radix: Int): String = this.toInt().toString(radix)\n\n/**r\n * Returns a string
representation of this [Short] value in the specified [radix].\n * @throws IllegalArgumentException when [radix] is not a valid
radix for number to string conversion.\n
*^r\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n//@kotlin.internal.InlineOnly\npublic /*inline*/
fun UShort.toString(radix: Int): String = this.toInt().toString(radix)\n\n\n/**r\n * Returns a string
representation of this [Int] value in the specified [radix].\n * @throws IllegalArgumentException when
[radix] is not a valid radix for number to string conversion.\n
*^r\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n//@kotlin.internal.InlineOnly\npublic /*inline*/
fun UInt.toString(radix: Int): String = this.toLong().toString(radix)\n\n\n/**r\n * Returns a string
representation of this [Long] value in the specified [radix].\n * @throws IllegalArgumentException when [radix] is not a valid
radix for number to string conversion.\n
*^r\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic
fun ULong.toString(radix: Int): String = ulongToString(this.toLong(), checkRadix(radix))\n\n\n\n/**r\n * Parses

```

the string as a signed [UByte] number and returns the result.
 @throws NumberFormatException if the string is not a valid representation of a number.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toUByte(): UByte = toUByteOrNull() ?: numberFormatError(this)
 Parses the string as a signed [UByte] number and returns the result.
 @throws NumberFormatException if the string is not a valid representation of a number.
 @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toUByte(radix: Int): UByte = toUByteOrNull(radix) ?: numberFormatError(this)
 Parses the string as a [UShort] number and returns the result.
 @throws NumberFormatException if the string is not a valid representation of a number.
 @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toUShort(): UShort = toUShortOrNull() ?: numberFormatError(this)
 Parses the string as a [UShort] number and returns the result.
 @throws NumberFormatException if the string is not a valid representation of a number.
 @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toUShort(radix: Int): UShort = toUShortOrNull(radix) ?: numberFormatError(this)
 Parses the string as an [UInt] number and returns the result.
 @throws NumberFormatException if the string is not a valid representation of a number.
 @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toInt(): UInt = toUIntOrNull() ?: numberFormatError(this)
 Parses the string as an [UInt] number and returns the result.
 @throws NumberFormatException if the string is not a valid representation of a number.
 @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toInt(radix: Int): UInt = toUIntOrNull(radix) ?: numberFormatError(this)
 Parses the string as a [ULong] number and returns the result.
 @throws NumberFormatException if the string is not a valid representation of a number.
 @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toULong(): ULong = toULongOrNull() ?: numberFormatError(this)
 Parses the string as a [ULong] number and returns the result.
 @throws NumberFormatException if the string is not a valid representation of a number.
 @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toULong(radix: Int): ULong = toULongOrNull(radix) ?: numberFormatError(this)
 Parses the string as an [UByte] number and returns the result
 or `null` if the string is not a valid representation of a number.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toUByteOrNull(): UByte? = toUByteOrNull(radix = 10)
 Parses the string as an [UByte] number and returns the result
 or `null` if the string is not a valid representation of a number.
 @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toUByteOrNull(radix: Int): UByte? {
 val int = this.toIntOrNull(radix) ?: return null
 if (int > UByte.MAX_VALUE) return null
 return int.toUByte()
 }
 Parses the string as an [UShort] number and returns the result
 or `null` if the string is not a valid representation of a number.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toUShortOrNull(): UShort? = toUShortOrNull(radix = 10)
 Parses the string as an [UShort] number and returns the result
 or `null` if the string is not a valid representation of a number.
 @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toUShortOrNull(radix: Int): UShort? {
 val int = this.toIntOrNull(radix) ?: return null
 if (int > UShort.MAX_VALUE) return null
 return int.toUShort()
 }
 Parses the string as an [UInt] number and returns the result
 or `null` if the string is not a valid representation of a number.
 @SinceKotlin("1.3")
 @ExperimentalUnsignedTypes
 public fun String.toUIntOrNull(): UInt? = toUIntOrNull(radix = 10)
 Parses the string as an [UInt] number and returns the result
 or `null` if the string is not a valid representation of a number.
 @throws IllegalArgumentException when [radix] is

not a valid radix for string to number conversion.

```
*@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun String.toIntOrNull(radix: Int): UInt?\n{\n    checkRadix(radix)\n    val length = this.length\n    if (length == 0) return null\n    val limit: UInt = UInt.MAX_VALUE\n    val start: Int\n    val firstChar = this[0]\n    if (firstChar < '0') {\n        if (length == 1 || firstChar != '+') return null\n        start = 1\n    } else {\n        start = 0\n    }\n    val uradix = radix.toInt()\n    val limitBeforeMul = limit / uradix\n    var result = 0u\n    for (i in start until length) {\n        val digit = digitOf(this[i], radix)\n        if (digit < 0) return null\n        if (result > limitBeforeMul) return null\n        result *= uradix\n        val beforeAdding = result\n        result += digit.toInt()\n        if (result < beforeAdding) return null // overflow has happened\n    }\n    return result\n}\n\n/* Parses the string as an [ULong] number and returns the result or `null` if the string is not a valid representation of a number.\n */\n\n*@\n@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun String.toULongOrNull(): ULong?\n= toULongOrNull(radix = 10)\n\n/* Parses the string as an [ULong] number and returns the result or `null` if the string is not a valid representation of a number.\n */\n\n*@\n@throws\nIllegalArgument\nException when [radix] is not a valid radix for string to number conversion.\n
```

```
*@\n@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun String.toULongOrNull(radix: Int): ULong?\n{\n    checkRadix(radix)\n    val length = this.length\n    if (length == 0) return null\n    val limit: ULong = ULong.MAX_VALUE\n    val start: Int\n    val firstChar = this[0]\n    if (firstChar < '0') {\n        if (length == 1 || firstChar != '+') return null\n        start = 1\n    } else {\n        start = 0\n    }\n    val uradix = radix.toInt()\n    val limitBeforeMul = limit / uradix\n    var result = 0uL\n    for (i in start until length) {\n        val digit = digitOf(this[i], radix)\n        if (digit < 0) return null\n        if (result > limitBeforeMul) return null\n        result *= uradix\n        val beforeAdding = result\n        result += digit.toInt()\n        if (result < beforeAdding) return null // overflow has happened\n    }\n    return result\n}\n\n/* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */\n
```

```
*@\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("MathKt")\npackage\nkotlin.math\n// constants, can't use them from nativeMath as they are not constants there\n/** Ratio of the circumference of a circle to its diameter, approximately 3.14159.\n */\n@SinceKotlin("1.2")\npublic const\nval PI: Double = 3.141592653589793\n/** Base of the natural logarithms, approximately 2.71828.\n */\n
```

```
*@\n@SinceKotlin("1.2")\npublic const\nval E: Double = 2.718281828459045\n// region\n\n==== Double Math =====\n\n/** Computes the sine of the angle [x] given in radians.\n */\n* Special cases:\n * - `sin(NaN|+Inf|-Inf)` is `NaN`\n\n*@\n@SinceKotlin("1.2")\npublic expect\nfun sin(x: Double): Double\n\n/** Computes the cosine of the angle [x] given in radians.\n */\n* Special cases:\n * - `cos(NaN|+Inf|-Inf)` is `NaN`\n\n*@\n@SinceKotlin("1.2")\npublic expect\nfun cos(x: Double): Double\n\n/** Computes the tangent of the angle [x] given in radians.\n */\n* Special cases:\n * - `tan(NaN|+Inf|-Inf)` is `NaN`\n\n*@\n@SinceKotlin("1.2")\npublic expect\nfun tan(x: Double): Double\n\n/** Computes the arc sine of the value [x];\n */\n* the returned value is an angle in the range from  $-\pi/2$  to  $\pi/2$  radians.\n */\n* Special cases:\n * - `asin(x)` is `NaN`, when  $abs(x) > 1$  or x is `NaN`\n\n*@\n@SinceKotlin("1.2")\npublic expect\nfun asin(x: Double): Double\n\n/** Computes the arc cosine of the value [x];\n */\n* the returned value is an angle in the range from  $0.0$  to  $\pi$  radians.\n */\n* Special cases:\n * - `acos(x)` is `NaN`, when  $abs(x) > 1$  or x is `NaN`\n\n*@\n@SinceKotlin("1.2")\npublic expect\nfun acos(x: Double): Double\n\n/** Computes the arc tangent of the value [x];\n */\n* the returned value is an angle in the range from  $-\pi/2$  to  $\pi/2$  radians.\n */\n* Special cases:\n * - `atan(NaN)` is `NaN`\n\n*@\n@SinceKotlin("1.2")\npublic expect\nfun atan(x: Double): Double\n\n/** Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond\n */\n* to the rectangular coordinates `(x, y)` by computing the arc tangent of the value  $y / x$ ;\n */\n* the returned value is an angle in the range from  $-\pi$  to  $\pi$  radians.\n */\n* Special cases:\n * - `atan2(0.0, 0.0)` is  $0.0$ \n */\n* - `atan2(0.0, x)` is  $0.0$  for  $x > 0$  and  $\pi$  for  $x < 0$ \n */\n* - `atan2(-0.0, x)` is  $-0.0$  for  $x > 0$  and  $-\pi$  for  $x < 0$ \n */\n* - `atan2(y, +Inf)` is  $0.0$  for  $0 < y < +Inf$  and  $-0.0$  for  $-\pi < y < 0$ \n */\n* - `atan2(y, -Inf)`\n
```

`is`PI` for $0 < y < +\text{Inf}$ and `-PI` for $-\text{Inf} < y < 0$ * -`atan2(y, 0.0)` is $\text{PI}/2$ for $y > 0$ and `-PI/2` for $y < 0$ * -`atan2(+Inf, x)` is $\text{PI}/2$ for finite x * -`atan2(-Inf, x)` is $-\text{PI}/2$ for finite x * -`atan2(NaN, x)` and `atan2(y, NaN)` is NaN *@SinceKotlin("1.2")\r\npublic expect fun atan2(y: Double, x: Double): Double\r\n\r\n/**\r\n * Computes the hyperbolic sine of the value [x].\r\n * Special cases:\r\n * -`sinh(NaN)` is NaN * -`sinh(+Inf)` is $+\text{Inf}$ * -`sinh(-Inf)` is $-\text{Inf}$ *@SinceKotlin("1.2")\r\npublic expect fun sinh(x: Double): Double\r\n\r\n/**\r\n * Computes the hyperbolic cosine of the value [x].\r\n * Special cases:\r\n * -`cosh(NaN)` is NaN * -`cosh(+Inf|-Inf)` is $+\text{Inf}$ *@SinceKotlin("1.2")\r\npublic expect fun cosh(x: Double): Double\r\n\r\n/**\r\n * Computes the hyperbolic tangent of the value [x].\r\n * Special cases:\r\n * -`tanh(NaN)` is NaN * -`tanh(+Inf)` is 1.0 * -`tanh(-Inf)` is -1.0 *@SinceKotlin("1.2")\r\npublic expect fun tanh(x: Double): Double\r\n\r\n/**\r\n * Computes the inverse hyperbolic sine of the value [x].\r\n * The returned value is `y` such that `sinh(y) == x` * Special cases:\r\n * -`asinh(NaN)` is NaN * -`asinh(+Inf)` is $+\text{Inf}$ * -`asinh(-Inf)` is $-\text{Inf}$ *@SinceKotlin("1.2")\r\npublic expect fun asinh(x: Double): Double\r\n\r\n/**\r\n * Computes the inverse hyperbolic cosine of the value [x].\r\n * The returned value is positive `y` such that `cosh(y) == x` * Special cases:\r\n * -`acosh(NaN)` is NaN * -`acosh(x)` is NaN when $x < 1$ * -`acosh(+Inf)` is $+\text{Inf}$ *@SinceKotlin("1.2")\r\npublic expect fun acosh(x: Double): Double\r\n\r\n/**\r\n * Computes the inverse hyperbolic tangent of the value [x].\r\n * The returned value is `y` such that `tanh(y) == x` * Special cases:\r\n * -`tanh(NaN)` is NaN * -`tanh(x)` is NaN when $x > 1$ or $x < -1$ * -`tanh(1.0)` is $+\text{Inf}$ * -`tanh(-1.0)` is $-\text{Inf}$ *@SinceKotlin("1.2")\r\npublic expect fun atanh(x: Double): Double\r\n\r\n/**\r\n * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.\r\n * Special cases:\r\n * - returns `+Inf` if any of arguments is infinite\r\n * - returns `NaN` if any of arguments is `NaN` and the other is not infinite\r\n *@SinceKotlin("1.2")\r\npublic expect fun hypot(x: Double, y: Double): Double\r\n\r\n/**\r\n * Computes the positive square root of the value [x].\r\n * Special cases:\r\n * -`sqrt(x)` is NaN when $x < 0$ or x is NaN \r\n *@SinceKotlin("1.2")\r\npublic expect fun sqrt(x: Double): Double\r\n\r\n/**\r\n * Computes Euler's number `e` raised to the power of the value [x].\r\n * Special cases:\r\n * -`exp(NaN)` is NaN * -`exp(+Inf)` is $+\text{Inf}$ * -`exp(-Inf)` is 0.0 *@SinceKotlin("1.2")\r\npublic expect fun exp(x: Double): Double\r\n\r\n/**\r\n * Computes `exp(x) - 1`.\r\n * This function can be implemented to produce more precise result for [x] near zero.\r\n * Special cases:\r\n * -`expm1(NaN)` is NaN * -`expm1(+Inf)` is $+\text{Inf}$ * -`expm1(-Inf)` is -1.0 *@see [exp] function.\r\n *@SinceKotlin("1.2")\r\npublic expect fun expm1(x: Double): Double\r\n\r\n/**\r\n * Computes the logarithm of the value [x] to the given [base].\r\n * Special cases:\r\n * -`log(x, b)` is NaN if either x or b are NaN * -`log(x, b)` is NaN when $x < 0$ or $b \leq 0$ or $b == 1.0$ * -`log(+Inf, +Inf)` is NaN * -`log(+Inf, b)` is $+\text{Inf}$ for $b > 1$ and $-\text{Inf}$ for $b < 1$ * -`log(0.0, b)` is $-\text{Inf}$ for $b > 1$ and $+\text{Inf}$ for $b < 1$ * See also logarithm functions for common fixed bases: [ln], [log10] and [log2].\r\n *@SinceKotlin("1.2")\r\npublic expect fun log(x: Double, base: Double): Double\r\n\r\n/**\r\n * Computes the natural logarithm (base `E`) of the value [x].\r\n * Special cases:\r\n * -`ln(NaN)` is NaN * -`ln(x)` is NaN when $x < 0.0$ * -`ln(+Inf)` is $+\text{Inf}$ * -`ln(0.0)` is $-\text{Inf}$ *@SinceKotlin("1.2")\r\npublic expect fun ln(x: Double): Double\r\n\r\n/**\r\n * Computes the common logarithm (base 10) of the value [x].\r\n * @see [ln] function for special cases.\r\n *@SinceKotlin("1.2")\r\npublic expect fun log10(x: Double): Double\r\n\r\n/**\r\n * Computes the binary logarithm (base 2) of the value [x].\r\n * @see [ln] function for special cases.\r\n *@SinceKotlin("1.2")\r\npublic expect fun log2(x: Double): Double\r\n\r\n/**\r\n * Computes `ln(x + 1)`.\r\n * This function can be implemented to produce more precise result for [x] near zero.\r\n * Special cases:\r\n * -`ln1p(NaN)` is NaN * -`ln1p(x)` is NaN where $x < -1.0$ * -`ln1p(-1.0)` is $-\text{Inf}$ * -`ln1p(+Inf)` is $+\text{Inf}$ *@see [ln] function\r\n * @see [expm1] function\r\n *@SinceKotlin("1.2")\r\npublic expect fun ln1p(x: Double): Double\r\n\r\n/**\r\n * Rounds the given value [x] to an integer towards positive infinity.\r\n * @return the smallest double value that is greater than or equal to the given value [x] and is a mathematical integer.\r\n * Special cases:\r\n * -`ceil(x)` is x where x is NaN`

or `+Inf` or `-Inf` or already a mathematical integer.

`public expect fun ceil(x: Double): Double`
Rounds the given value [x] to an integer towards negative infinity.
@return the largest double value that is smaller than or equal to the given value [x] and is a mathematical integer.
Special cases: `-floor(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.

`public expect fun floor(x: Double): Double`
Rounds the given value [x] to an integer towards zero.
@return the value [x] having its fractional part truncated.
Special cases: `-truncate(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.

`public expect fun truncate(x: Double): Double`
Rounds the given value [x] towards the closest integer with ties rounded towards even integer.
Special cases: `-round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.

`public expect fun round(x: Double): Double`
Returns the absolute value of the given value [x].
Special cases: `-abs(NaN)` is `NaN`.
@see `absoluteValue` extension property for [Double].

`public expect fun abs(x: Double): Double`
Returns the sign of the given value [x]:
`-1.0` if the value is negative,
zero if the value is zero,
`1.0` if the value is positive.
Special case: `-sign(NaN)` is `NaN`.

`public expect fun sign(x: Double): Double`
Returns the smaller of two values.
If either value is `NaN`, then the result is `NaN`.

`public expect fun min(a: Double, b: Double): Double`
Returns the greater of two values.
If either value is `NaN`, then the result is `NaN`.

`public expect fun max(a: Double, b: Double): Double`
// extensions

`public expect fun Double.pow(x: Double): Double`
Raises this value to the power [x].
Special cases: `-b.pow(0.0)` is `1.0`
`-b.pow(1.0) == b`
`-b.pow(NaN)` is `NaN`
`-NaN.pow(x)` is `NaN` for `x != 0.0`
`-b.pow(Inf)` is `NaN` for `abs(b) == 1.0`
`-b.pow(x)` is `NaN` for `b < 0` and `x` is finite and not an integer.

`public expect fun Double.pow(x: Double): Double`
Raises this value to the integer power [n].
See the other overload of [pow] for details.

`public expect fun Double.pow(n: Int): Double`
Returns the absolute value of this value.
Special cases: `-NaN.absoluteValue` is `NaN`.
@see `abs` function.

`public expect val Double.absoluteValue: Double`
Returns the sign of this value:
`-1.0` if the value is negative,
zero if the value is zero,
`1.0` if the value is positive.
Special case: `-NaN.sign` is `NaN`.

`public expect val Double.sign: Double`
Returns this value with the sign bit same as of the [sign] value.
If [sign] is `NaN` the sign of the result is undefined.

`public expect fun Double.withSign(sign: Double): Double`
Returns this value with the sign bit same as of the [sign] value.

`public expect fun Double.withSign(sign: Int): Double`
Returns the ulp (unit in the last place) of this value.
An ulp is a positive distance between this value and the next nearest [Double] value larger in magnitude.
Special Cases: `-NaN.ulp` is `NaN`
`-x.ulp` is `+Inf` when `x` is `+Inf` or `-Inf`
`-0.0.ulp` is `Double.MIN_VALUE`.

`public expect val Double.ulp: Double`
Returns the [Double] value nearest to this value in direction of positive infinity.

`public expect fun Double.nextUp(): Double`
Returns the [Double] value nearest to this value in direction of negative infinity.

`public expect fun Double.nextDown(): Double`
Returns the [Double] value nearest to this value in direction from this value towards the value [to].
Special cases: `-x.nextTowards(y)` is `NaN` if either `x` or `y` are `NaN`
`-x.nextTowards(x) == x`.

`public expect fun Double.nextTowards(to: Double): Double`
Rounds this [Double] value to the nearest integer and converts the result to [Int].
Ties are rounded towards positive infinity.
Special cases: `-x.roundToInt() == Int.MAX_VALUE` when `x > Int.MAX_VALUE`
`-x.roundToInt() == Int.MIN_VALUE` when `x < Int.MIN_VALUE`.
@throws `IllegalArgumentException` when this value is `NaN`.

`public expect fun`

`Double.roundToInt(): Int` Rounds this [Double] value to the nearest integer and converts the result to [Long]. Ties are rounded towards positive infinity. Special cases: `-x.roundToLong() == Long.MAX_VALUE` when `x > Long.MAX_VALUE` `-x.roundToLong() == Long.MIN_VALUE` when `x < Long.MIN_VALUE` @throws `IllegalArgumentException` when this value is `NaN`

`public expect fun Double.roundToLong(): Long` // endregion

`Float Math`

`Computes the sine of the angle [x] given in radians.` Special cases: `-sin(NaN|+Inf|-Inf)` is `NaN`

`public expect fun sin(x: Float): Float` Computes the cosine of the angle [x] given in radians. Special cases: `-cos(NaN|+Inf|-Inf)` is `NaN`

`public expect fun cos(x: Float): Float` Computes the tangent of the angle [x] given in radians. Special cases: `-tan(NaN|+Inf|-Inf)` is `NaN`

`public expect fun tan(x: Float): Float` Computes the arc sine of the value [x]; the returned value is an angle in the range from `-PI/2` to `PI/2` radians. Special cases: `-asin(x)` is `NaN`, when `abs(x) > 1` or `x` is `NaN`

`public expect fun asin(x: Float): Float` Computes the arc cosine of the value [x]; the returned value is an angle in the range from `0.0` to `PI` radians. Special cases: `-acos(x)` is `NaN`, when `abs(x) > 1` or `x` is `NaN`

`public expect fun acos(x: Float): Float` Computes the arc tangent of the value [x]; the returned value is an angle in the range from `-PI/2` to `PI/2` radians. Special cases: `-atan(NaN)` is `NaN`

`public expect fun atan(x: Float): Float` Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond to the rectangular coordinates `(x, y)` by computing the arc tangent of the value `y / x`; the returned value is an angle in the range from `-PI` to `PI` radians. Special cases: `-atan2(0.0, 0.0)` is `0.0` `-atan2(0.0, x)` is `0.0` for `x > 0` and `PI` for `x < 0` `-atan2(-0.0, x)` is `-0.0` for `x > 0` and `-PI` for `x < 0` `-atan2(y, +Inf)` is `0.0` for `0 < y < +Inf` and `-0.0` for `-Inf < y < 0` `-atan2(y, -Inf)` is `PI` for `0 < y < +Inf` and `-PI` for `-Inf < y < 0` `-atan2(y, 0.0)` is `PI/2` for `y > 0` and `-PI/2` for `y < 0` `-atan2(+Inf, x)` is `PI/2` for finite `x` `-atan2(-Inf, x)` is `-PI/2` for finite `x` `-atan2(NaN, x)` and `-atan2(y, NaN)` is `NaN`

`public expect fun atan2(y: Float, x: Float): Float`

`Computes the hyperbolic sine of the value [x].` Special cases: `-sinh(NaN)` is `NaN` `-sinh(+Inf)` is `+Inf` `-sinh(-Inf)` is `-Inf`

`public expect fun sinh(x: Float): Float`

`Computes the hyperbolic cosine of the value [x].` Special cases: `-cosh(NaN)` is `NaN` `-cosh(+Inf|-Inf)` is `+Inf`

`public expect fun cosh(x: Float): Float`

`Computes the hyperbolic tangent of the value [x].` Special cases: `-tanh(NaN)` is `NaN` `-tanh(+Inf)` is `1.0` `-tanh(-Inf)` is `-1.0`

`public expect fun tanh(x: Float): Float`

`Computes the inverse hyperbolic sine of the value [x].` The returned value is `y` such that `sinh(y) == x`. Special cases: `-asinh(NaN)` is `NaN` `-asinh(+Inf)` is `+Inf` `-asinh(-Inf)` is `-Inf`

`public expect fun asinh(x: Float): Float`

`Computes the inverse hyperbolic cosine of the value [x].` The returned value is positive `y` such that `cosh(y) == x`. Special cases: `-acosh(NaN)` is `NaN` `-acosh(x)` is `NaN` when `x < 1` `-acosh(+Inf)` is `+Inf`

`public expect fun acosh(x: Float): Float`

`Computes the inverse hyperbolic tangent of the value [x].` The returned value is `y` such that `tanh(y) == x`. Special cases: `-tanh(NaN)` is `NaN` `-tanh(x)` is `NaN` when `x > 1` or `x < -1` `-tanh(1.0)` is `+Inf` `-tanh(-1.0)` is `-Inf`

`public expect fun atanh(x: Float): Float`

`Computes sqrt(x^2 + y^2) without intermediate overflow or underflow.` Special cases: returns `+Inf` if any of arguments is infinite returns `NaN` if any of arguments is `NaN` and the other is not infinite

`public expect fun hypot(x: Float, y: Float): Float`

`Computes the positive square root of the value [x].` Special cases: `-sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN`

`public expect fun sqrt(x: Float): Float`

Float Computes Euler's number e raised to the power of the value [x]. Special cases: $\exp(\text{NaN})$ is NaN , $\exp(+\text{Inf})$ is $+\text{Inf}$, $\exp(-\text{Inf})$ is 0.0 .

```
@SinceKotlin("1.2")
public expect fun exp(x: Float): Float
```

Computes $\exp(x) - 1$. This function can be implemented to produce more precise result for [x] near zero. Special cases: $\expm1(\text{NaN})$ is NaN , $\expm1(+\text{Inf})$ is $+\text{Inf}$, $\expm1(-\text{Inf})$ is -1.0 . @see [exp] function.

```
@SinceKotlin("1.2")
public expect fun expm1(x: Float): Float
```

Computes the logarithm of the value [x] to the given [base]. Special cases: $\log(x, b)$ is NaN if either x or b are NaN , $\log(x, b)$ is NaN when $x < 0$ or $b \leq 0$ or $b == 1.0$, $\log(+\text{Inf}, +\text{Inf})$ is NaN , $\log(+\text{Inf}, b)$ is $+\text{Inf}$ for $b > 1$ and $-\text{Inf}$ for $b < 1$, $\log(0.0, b)$ is $-\text{Inf}$ for $b > 1$ and $+\text{Inf}$ for $b > 1$. See also logarithm functions for common fixed bases: [ln], [log10] and [log2].

```
@SinceKotlin("1.2")
public expect fun log(x: Float, base: Float): Float
```

Computes the natural logarithm (base E) of the value [x]. Special cases: $\ln(\text{NaN})$ is NaN , $\ln(x)$ is NaN when $x < 0.0$, $\ln(+\text{Inf})$ is $+\text{Inf}$, $\ln(0.0)$ is $-\text{Inf}$.

```
@SinceKotlin("1.2")
public expect fun ln(x: Float): Float
```

Computes the common logarithm (base 10) of the value [x]. @see [ln] function for special cases.

```
@SinceKotlin("1.2")
public expect fun log10(x: Float): Float
```

Computes the binary logarithm (base 2) of the value [x]. @see [ln] function for special cases.

```
@SinceKotlin("1.2")
public expect fun log2(x: Float): Float
```

Computes $\ln(a + 1)$. This function can be implemented to produce more precise result for [x] near zero. Special cases: $\ln1p(\text{NaN})$ is NaN , $\ln1p(x)$ is NaN where $x < -1.0$, $\ln1p(-1.0)$ is $-\text{Inf}$, $\ln1p(+\text{Inf})$ is $+\text{Inf}$. @see [ln] function @see [expm1] function.

```
@SinceKotlin("1.2")
public expect fun ln1p(x: Float): Float
```

Rounds the given value [x] to an integer towards positive infinity. @return the smallest Float value that is greater than or equal to the given value [x] and is a mathematical integer. Special cases: $\text{ceil}(x)$ is x where x is NaN or $+\text{Inf}$ or $-\text{Inf}$ or already a mathematical integer.

```
@SinceKotlin("1.2")
public expect fun ceil(x: Float): Float
```

Rounds the given value [x] to an integer towards negative infinity. @return the largest Float value that is smaller than or equal to the given value [x] and is a mathematical integer. Special cases: $\text{floor}(x)$ is x where x is NaN or $+\text{Inf}$ or $-\text{Inf}$ or already a mathematical integer.

```
@SinceKotlin("1.2")
public expect fun floor(x: Float): Float
```

Rounds the given value [x] to an integer towards zero. @return the value [x] having its fractional part truncated. Special cases: $\text{truncate}(x)$ is x where x is NaN or $+\text{Inf}$ or $-\text{Inf}$ or already a mathematical integer.

```
@SinceKotlin("1.2")
public expect fun truncate(x: Float): Float
```

Rounds the given value [x] towards the closest integer with ties rounded towards even integer. Special cases: $\text{round}(x)$ is x where x is NaN or $+\text{Inf}$ or $-\text{Inf}$ or already a mathematical integer.

```
@SinceKotlin("1.2")
public expect fun round(x: Float): Float
```

Returns the absolute value of the given value [x]. Special cases: $\text{abs}(\text{NaN})$ is NaN . @see absoluteValue extension property for [Float].

```
@SinceKotlin("1.2")
public expect fun abs(x: Float): Float
```

Returns the sign of the given value [x]: -1.0 if the value is negative, zero if the value is zero, 1.0 if the value is positive. Special case: $\text{sign}(\text{NaN})$ is NaN .

```
@SinceKotlin("1.2")
public expect fun sign(x: Float): Float
```

Returns the smaller of two values. If either value is NaN , then the result is NaN .

```
@SinceKotlin("1.2")
public expect fun min(a: Float, b: Float): Float
```

Returns the greater of two values. If either value is NaN , then the result is NaN .

```
@SinceKotlin("1.2")
public expect fun max(a: Float, b: Float): Float
```

Raises this value to the power [x]. Special cases: $b.\text{pow}(0.0)$ is 1.0 , $b.\text{pow}(1.0) == b$, $b.\text{pow}(\text{NaN})$ is NaN , $\text{NaN}.\text{pow}(x)$ is NaN for $x \neq 0.0$, $b.\text{pow}(\text{Inf})$ is NaN for $\text{abs}(b) == 1.0$, $b.\text{pow}(x)$ is NaN for $b < 0$ and x is finite and not an integer.

```
@SinceKotlin("1.2")
public expect fun Float.pow(x: Float): Float
```

Raises this value to the integer power [n]. See the other overload of [pow] for details.

```
@SinceKotlin("1.2")
public expect fun Float.pow(n: Int): Float
```

Returns the absolute value

of this value.
Special cases: `- NaN.absoluteValue` is `- NaN`
@see `abs` function
@SinceKotlin("1.2")
public expect val `Float.absoluteValue`: `Float`
Returns the sign of this value:
- `-1.0` if the value is negative,
- zero if the value is zero,
- `1.0` if the value is positive
Special case: `- NaN.sign` is `- NaN`
@SinceKotlin("1.2")
public expect val `Float.sign`: `Float`
Returns this value with the sign bit same as of the [sign] value.
If [sign] is `- NaN` the sign of the result is undefined.
@SinceKotlin("1.2")
public expect fun `Float.withSign`(sign: `Float`): `Float`
Returns this value with the sign bit same as of the [sign] value.
@SinceKotlin("1.2")
public expect fun `Float.withSign`(sign: `Int`): `Float`
Rounds this [Float] value to the nearest integer and converts the result to [Int].
Ties are rounded towards positive infinity.
Special cases: `- x.roundToInt() == Int.MAX_VALUE` when `x > Int.MAX_VALUE`
- `x.roundToInt() == Int.MIN_VALUE` when `x < Int.MIN_VALUE`
@throws `IllegalArgumentException` when this value is `- NaN`
@SinceKotlin("1.2")
public expect fun `Float.roundToInt`(): `Int`
Rounds this [Float] value to the nearest integer and converts the result to [Long].
Ties are rounded towards positive infinity.
Special cases: `- x.roundToLong() == Long.MAX_VALUE` when `x > Long.MAX_VALUE`
- `x.roundToLong() == Long.MIN_VALUE` when `x < Long.MIN_VALUE`
@throws `IllegalArgumentException` when this value is `- NaN`
@SinceKotlin("1.2")
public expect fun `Float.roundToLong`(): `Long`
endregion
region
===== Integer Math =====
Returns the absolute value of the given value [n].
Special cases: `- abs(Int.MIN_VALUE)` is `- Int.MIN_VALUE` due to an overflow
@see `absoluteValue` extension property for [Int]
@SinceKotlin("1.2")
public expect fun `abs`(n: `Int`): `Int`
Returns the smaller of two values.
@SinceKotlin("1.2")
public expect fun `min`(a: `Int`, b: `Int`): `Int`
Returns the greater of two values.
@SinceKotlin("1.2")
public expect fun `max`(a: `Int`, b: `Int`): `Int`
Returns the absolute value of this value.
Special cases: `- Int.MIN_VALUE.absoluteValue` is `- Int.MIN_VALUE` due to an overflow
@see `abs` function
@SinceKotlin("1.2")
public expect val `Int.absoluteValue`: `Int`
Returns the sign of this value:
- `-1` if the value is negative,
- `0` if the value is zero,
- `1` if the value is positive
@SinceKotlin("1.2")
public expect val `Int.sign`: `Int`
Returns the absolute value of the given value [n].
Special cases: `- abs(Long.MIN_VALUE)` is `- Long.MIN_VALUE` due to an overflow
@see `absoluteValue` extension property for [Long]
@SinceKotlin("1.2")
public expect fun `abs`(n: `Long`): `Long`
Returns the smaller of two values.
@SinceKotlin("1.2")
public expect fun `min`(a: `Long`, b: `Long`): `Long`
Returns the greater of two values.
@SinceKotlin("1.2")
public expect fun `max`(a: `Long`, b: `Long`): `Long`
Returns the absolute value of this value.
Special cases: `- Long.MIN_VALUE.absoluteValue` is `- Long.MIN_VALUE` due to an overflow
@see `abs` function
@SinceKotlin("1.2")
public expect val `Long.absoluteValue`: `Long`
Returns the sign of this value:
- `-1` if the value is negative,
- `0` if the value is zero,
- `1` if the value is positive
@SinceKotlin("1.2")
public expect val `Long.sign`: `Int`
endregion
",null,null,null,null,null,null,null,null], "names": [], "mappings": "AAWC,0B;EAVG,IAAI,OAAO,MAAO,KAAl,UAAW,IAAG,MAAM,IAA1C,C;IACI,MAAM,CAAC,QAAD,EAAW,CAAC,SAAD,CAAX,EAAwB,OAAxB,C;;QAEI,IAAI,OAAO,OAAQ,KAAl,QAAvB,C;IACD,OAAO,CAAC,MAAM,QAAP,C;;IAGP,IAAI,OAAQ,GAAE,E;IACd,OAAO,CAAC,IAAI,OAAL,C;;AAEf,CAAC,CAAC,IAAD,oB;EACG,IAAI,IAAI,M;ECPZ,MAAM,eAAgB,gB;IACIB,OAAoD,CAA5C,KAAK,QAAQ,CAAC,CAAD,CAAl,IAAG,CAAe,YAAW,SAAW,KAAAG,CAAC,OAAQ,KAAl,c;EACxE,C;EAEA,MAAM,YAAa,gB;IACf,OAAO,CAAe,YAAW,SAAU,IAAG,CAAC,OAAQ,KAAl,c;EACID,C;EAEA,MAAM,aAAc,gB;IACbB,OAAO,CAAe,YAAW,U;EACxB,C;EAEA,MAAM,YAAa,gB;IACf,OAAO,CAAe,YAAW,WAAY,IAAG,CAAC,OAAQ,KAAl,W;EACpD,C;EAEA,MAAM,WAAY,gB;IACd,OAAO,CAAe,YAAW,U;EACxB,C;EAEA,MAAM,aAAc,gB;IACbB,OAAO,CAAe,YAAW,Y;EACxB,C;EAEA,MAAM,cAAe,gB;IACjB,OAAO,CAAe,YAAW,Y;EACxB,C;EAEA,MAAM,YAAa,gB;IACf,OAAO,KAAK,QAAQ,CAAC,CAAD,CAAl,IAAG,CAAC,OAAQ,KAAl,W;EAC5C,C;EAEA,MAAM,QAAS,gB;IACX,OAA

O, KAAK, QAAQ, CAAC, CAAD, CAAL, IAAG, CAAC, CAAC, O; EACjC, C; EAEA, MAAM, WAAY, gB; IACd, OAAO, KAAK, QAAQ, CAAC, CAAD, CAAL, IAAG, WAAW, OAAO, CAAC, CAAD, C; EACjD, C; EAEA, MAAM, cAAe, gB; IACjB, IAAI, WAAW, MAAM, YAAY, CAAC, CAAD, CAAL, GAAE, MAAM, aAAR, GAAwB, MAAM, S; IACnE, OAAO, GAAI, GAAE, KAAK, UAAU, IAAI, KAAK, CAAC, CAAD, e; MAAkB, OAAO, QAAQ, CAAC, CAAD, C; IAAK, CAAtC, CAAwC, KAAK, CAAC, IAAD, CAAO, GAAE, G; EAC/F, C; EAEA, MAAM, kBAAmB, kB; IACrB, OAAO, MAAM, OAAO, YAAY, wBAAwB, CAAC, GAAD, C; EAC5D, C; EAEA, MAAM, YAAa, mB; IACf, IAAI, CAAE, KAAI, CA AV, C; MACI, OAAO, I; IAEX, IAAI, CAAC, MAAM, WAAW, CAAC, CAAD, CAAL, IAAG, CAAC, OAAQ, KAAI, CAAC, OAA3C, C; MACI, OAAO, K; IAGX, KAAK, IAAI, IAAI, CAAR, EAAW, IAAI, CAAC, OAArB, EAA8B, CAAE, GAAE, CAAIC, EAAqC, CAAC, EAAtC, C; MACI, IAAI, CAAC, MAAM, OAAO, CAAC, CAAC, CAAC, CAAD, CAAF, EAAO, CAAC, CAAC, CAAD, CAAR, CAAIB, C; QACI, OAAO, K; IAGf, OAAO, I; EACX, C; EAEA, MAAM, gBAAiB, mB; IACnB, OAAO, MAAM, OAAO, YAAY, sBAAsB, CAAC, CAAD, EAAI, CAAJ, C; EAC1D, C; EAEA, MAAM, cAAe, kB; IACjB, IAAI, SAAS, C; IACb, KAAK, IAAI, IAAI, CAAR, EAAW, IAAI, GAAG, OAAvB, EAAgC, CAAE, GA AE, CAApC, EAAuC, CAAC, EAAxC, C; MACI, MAAO, GAAqB, CAAjB, EAAG, GAAE, MAAO, GAAE, CAAG, IAA E, MAAM, SAAS, CAAC, GAAG, CAAC, CAAD, CAAJ, CAU, GAAE, C; IAE7D, OAAO, M; EACX, C; EAEA, MAAM, kBAAmB, kB; IACrB, OAAO, MAAM, OAAO, YAAY, wBAAwB, CAAC, GAAD, C; EAC5D, C; EAEA, MAAM, mBAAoB, oB; IACtB, KAAK, KAAK, CAAC, MAAM, gBAAP, C; EACd, C; ECnFA, MAAM, eAAgB, sB; IACIB, CAAC, aA Ac, GAAE, I; IACjB, OAAO, C; EACX, C; EAEA, MAAM, uBAAwB, +C; IAC1B, MAAM, IAAK, GAAE, M; IACb, MAAM, IAAK, GAAE, M; IACb, MAAM, aAAc, GAAE, I; IACtB, OAAO, mBAAmB, CAAC, MAAD, EAAS, MAAT, EAAiB, 6BAA6B, CAAC, UAAD, CAA9C, C; EAC9B, C; IAGI, GAAG, WAAY, GAAE, sBAAsB, CAAC, OAAO, MAAO, KAA I, UAAW, GAAE, KAAK, QAAP, GAAkB, KAAK, UAARd, C; IACvC, GAAG, YAAa, GAAE, G; IACIB, OAAO, G; EAC X, C; EAEA, IAAI, gCAAgC, CAChC, UACa, QAAS, IAAT, oC; IACL, OAAO, MAAM, OAAO, QAAQ, kB; EAAmB, CA DIC, CADb, aAIe, QAAS, IAAT, oC; IACP, OAAO, MAAM, OAAO, QAAQ, W; EAAy, CADjC, CAJf, CADgC, EAShC, UACa, QAAS, IAAT, oC; IACL, OAAO, MAAM, OAAO, QAAQ, kB; EAAmB, CAD1C, CADb, aAIe, QAAS, IAAT, oC; I ACP, OAAO, MAAM, OAAO, QAAQ, W; EAAy, CADjC, CAJf, CATgC, C; IAoBhC, IAAI, KAAK, MAAO, KAAI, IAA pB, C; MACI, KAAK, MAAO, GAAE, aACE, CAAC, KAAK, qBAAqB, EAA3B, CADF, aAEC, IAFD, aAGC, EAHD, cAI E, EAJF, SAKH, EALG, iBAMK, EANL, C; IASIB, OAAO, KAAK, M; EAChB, C; ECjDA, MAAM, QAAS, gB; IACX, O AAoB, CAAZ, CAAE, GAAE, KAAQ, KAAG, EAAG, IAAG, E; EACjC, C; EAEA, MAAM, OAAQ, gB; IACV, OAAkB, CAAV, CAAE, GAAE, GAAM, KAAG, EAAG, IAAG, E; EAC/B, C; EAEA, MAAM, OAAQ, gB; IACV, OAAO, CAAE, GAAE, K; EACf, C; EAEA, MAAM, aAAc, gB; IACb, OAAO, CAAE, YAAW, MAAM, KAAM, GAAE, CAAF, GAAM, MAAM, KAAK, WAAW, CAAC, CAAD, C; EAChE, C; EAEA, MAAM, YAAa, gB; IACf, OAAO, CAAE, YAAW, MAAM, KAAM, GAAE, CAAC, MAAM, EAAT, GAAC, MAAM, YAAY, CAAC, CAAD, C; EACpE, C; EAEA, MAAM, cAAe, gB; IACjB, OAAO, MAAM, QAAQ, CAAC, MAAM, YAAY, CAAC, CAAD, CAAAnB, C; EACzB, C; EAEA, MAAM, aA Ac, gB; IACb, OAAO, MAAM, OAAO, CAAC, MAAM, YAAY, CAAC, CAAD, CAAAnB, C; EACxB, C; EAEA, MAAM, eAAgB, gB; IACIB, OAAO, CAAC, C; EACZ, C; EAEA, MAAM, aAAc, gB; IACb, OAAO, MAAM, OAAO, CAAC, MAAM, YAAY, CAAC, CAAD, CAAAnB, C; EACxB, C; EAEA, MAAM, YAAa, gB; IACf, IAAI, CAAE, GAAE, UAAR, C; M AAoB, OAAO, U; IAC3B, IAAI, CAAE, GAAE, WAAR, C; MAAqB, OAAO, W; IAC5B, OAAO, CAAE, GAAE, C; EACf, C; EAEA, MAAM, YAAa, gB; IACf, IAAI, CAAE, IAAG, IAAT, C; MAAe, OAAO, C; IACtB, IAAI, CAAE, YAAW, MAAM, UAAvB, C; MAAmC, OAAO, C; IAC1C, OAAO, IAAI, MAAM, UAAV, CAAqB, CAARb, C; EACX, C; EAEA, MAAM, UAAW, gB; IACb, IAAI, CAAE, IAAG, IAAT, C; MAAe, OAAO, C; IACtB, OAAO, MAAM, OAAO, CAAC, CAAD, C; EACxB, C; ECnDA, MAAM, OAAQ, yB; IACV, IAAI, IAAK, IAAG, IAAZ, C; MACI, OAAO, IAAK, IAAG, I; IAGnB, I AAI, IAAK, IAAG, IAAZ, C; MACI, OAAO, K; IAGX, IAAI, IAAK, KAAI, IAAb, C; MACI, OAAO, IAAK, KAAI, I; IAG pB, IAAI, OAAO, IAAK, KAAI, QAAS, IAAG, OAAO, IAAI, OAAQ, KAAI, UAAvD, C; MACI, OAAO, IAAI, OAAO, C AAC, IAAD, C; IAGtB, IAAI, OAAO, IAAK, KAAI, QAAS, IAAG, OAAO, IAAK, KAAI, QAAd, C; MACI, OAAO, IA AK, KAAI, IAAK, KAAI, IAAK, KAAI, CAAE, IAAG, CAAE, GAAE, IAAK, KAAI, CAAE, GAAE, IAAAnC, C; IAGzB, OAAO, IAAK, KAAI, I; EACpB, C; EAEA, MAAM, SAAU, kB; IACZ, IAAI, GAAI, IAAG, IAAX, C; MACI, OAAO, C; I AEX, IAAI, UAAU, OAAO, G; IACrB, IAAI, QAAS, KAAI, OAAjB, C; MACI, OAAO, UAAW, KAAI, OAAO, GAAG, S AAU, GAAE, GAAG, SAAS, EAAd, GAAMb, iBAAiB, CAAC, GAAD, C; IAEIF, IAAI, UAAW, KAAI, OAAAnB, C; MA CI, OAAO, iBAAiB, CAAC, GAAD, C; IAE5B, IAAI, QAAS, KAAI, OAAjB, C; MACI, OAAO, MAAM, eAAe, CAAC, G

AAD,C;;IAEhC,IAAI,SAAU,KAAI,OAAiB,C;MACI,OAAO,MAAM,CAAC,GAAD,C;;IAGjB,IAAI,MAAM,MAAM,CAAC,GAAD,C;IACHB,OAAO,iBAAiB,CAAC,GAAD,C;EAC5B,C;EAGA,MAAM,SAAU,gB;IACZ,IAAI,CAAE,IAAG,IAAT,C;MACI,OAAO,M;;UAEN,IAAI,MAAM,WAAW,CAAC,CAAD,CAArB,C;MACD,OAAO,O;;;MAGP,OAAO,CAAC,SAAS,E;;EAEzB,C;EAGA,IAAI,WAAW,a;EAGf,IAAI,iCAAiC,sB;;IAGjC,IAAI,EAAE,8BAA+B,IAAG,GAAPC,CAAJ,C;MACI,IAAI,OAAQ,IAAI,OAAO,EAAG,GAAE,QAAU,GAAE,C;MACxC,MAAM,eAAe,CAAC,GAAD,EAAM,8BAAN,EAAsC,QAAU,IAAV,cAA4B,KAA5B,CAAtC,C;;IAEzB,OAAO,GAA G,CAAC,8BAAD,C;EACd,C;;IAGI,IAAI,OAAO,C;IACX,KAAK,IAAI,IAAI,CAAb,EAAGB,CAAE,GAAE,GAA G,OAAvB,EAAGC,CAAC,EAajC,C;MACI,IAAI,OAAQ,GAAG,WAAW,CAAC,CAAD,C;MAC1B,IAAM,GAA G,IAAK,GAAE,EAAG,GAAE,IAAM,GAAE,C;;IAEjC,OAAO,I;EACX,C;EAEA,MAAM,iBAaKB,GAAE,iB;EC9 C1B,MAAM,KAAM,wB;IAKV,IAAI,KAAM,GAAE,GAAI,GAAE,C;IAMIB,IAAI,MAAO,GAAE,IAAK,GAAE, C;EACtB,C;EAEA,MAAM,KAAK,WAA Y,GAAE,OACf,OADe,cAET,MAFS,cAGV,EAHU,C;EA gBzB,MAAM, KAAK,UAAW,GAAE,E;EAQxB,MAAM,KAAK,QAAS,oB;IACIB,IAAI,IAAK,IAAG,KAAM,IAAG,KAAM,GA AE,GAA7B,C;MACE,IAAI,YAAY,MAAM,KAAK,UAAU,CAAC,KAAD,C;MACrC,IAAI,SAAJ,C;QACE,OAA O,S;;;IAIX,IAAI,MAAM,IAAI,MAAM,KAAV,CAAgB,KAAM,GAAE,CAAxB,EAA2B,KAAM,GAAE,CAAE,G AAE,EAAF,GAAO,CAA5C,C;IACV,IAAI,IAAK,IAAG,KAAM,IAAG,KAAM,GAAE,GAA7B,C;MACE,MAAM ,KAAK,UAAU,CAAC,KAAD,CAAQ,GAAE,G;;IAEjC,OAAO,G;EACT,C;EAWA,MAAM,KAAK,WAA Y,oB;IA CrB,IAAI,KAAK,CAAC,KAAD,CAAT,C;MACE,OAAO,MAAM,KAAK,K;;UACb,IAAI,KAAM,IAAG,CAAC, MAAM,KAAK,gBAazB,C;MACL,OAAO,MAAM,KAAK,U;;UACb,IAAI,KAAM,GAAE,CAAE,IAAG,MAAM, KAAK,gBAA5B,C;MACL,OAAO,MAAM,KAAK,U;;UACb,IAAI,KAAM,GAAE,CAAZ,C;MACL,OAAO,MAA M,KAAK,WAAW,CAAC,CAAC,KAAF,CAAQ,OAAO,E;;;MAE5C,OAAO,IAAI,MAAM,KAAV,CACF,KAAM, GAAE,MAAM,KAAK,gBAaKB,GAAE,CADrC,EAEF,KAAM,GAAE,MAAM,KAAK,gBAaKB,GAAE,CAFrC,C ;;EAIX,C;EAUA,MAAM,KAAK,SAAU,gC;IACnB,OAAO,IAAI,MAAM,KAAV,CAAgB,OAAhB,EAAYB,QAaz B,C;EACT,C;EAUA,MAAM,KAAK,WAA Y,6B;IACrB,IAAI,GAAG,OAAQ,IAAG,CAAiB,C;MACE,MAAM,KA AK,CAAC,mCAAD,C;;IAGb,IAAI,QAAQ,SAAU,IAAG,E;IACzB,IAAI,KAAM,GAAE,CAAE,IAAG,EAAG,GA AE,KAAtB,C;MACE,MAAM,KAAK,CAAC,sBAAuB,GAAE,KAA1B,C;;IAGb,IAAI,GAAG,OAAO,CAAC,CAA D,CAAI,IAAG,GAArB,C;MACE,OAAO,MAAM,KAAK,WAAW,CAAC,GAAG,UAAU,CAAC,CAAD,CAAd,E AAmB,KAA nB,CAAYB,OAAO,E;;UACxD,IAAI,GAAG,QAAQ,CAAC,GAAD,CAAM,IAAG,CAAxB,C;MACL, MAAM,KAAK,CAAC,+CAAGD,GAAE,GAAnD,C;;IAKb,IAAI,eAAe,MAAM,KAAK,WAAW,CAAC,IAAI,IAA I,CAAC,KAAD,EAAQ,CAAR,CAAT,C;IAEzC,IAAI,SAAS,MAAM,KAAK,K;IACxB,KAAK,IAAI,IAAI,CAAb, EAAGB,CAAE,GAAE,GAAG,OAAvB,EAAGC,CAAE,IAAG,CAArC,C;MACE,IAAI,OAAO,IAAI,IAAI,CAAC,C AAD,EAAI,GAAG,OAAQ,GAAE,CAAjB,C;MACnB,IAAI,QAAQ,QAAQ,CAAC,GAAG,UAAU,CAAC,CAAD, EAAI,CAAE,GAAE,IAAR,CAAd,EAA6B,KAA7B,C;MACpB,IAAI,IAAK,GAAE,CAAX,C;QACE,IAAI,QAAQ, MAAM,KAAK,WAAW,CAAC,IAAI,IAAI,CAAC,KAAD,EAAQ,IAAR,CAAT,C;QACIC,MAAO,GAAE,MAAM ,SAAS,CAAC,KAAD,CAAO,IAAI,CAAC,MAAM,KAAK,WAAW,CAAC,KAAD,CAAvB,C;;;QAE nC,MAAO,G AAE,MAAM,SAAS,CAAC,YAAD,C;QACxB,MAAO,GAAE,MAAM,IAAI,CAAC,MAAM,KAAK,WAAW,CAA C,KAAD,CAAvB,C;;;IAGvB,OAAO,M;EACT,C;EAaA,MAAM,KAAK,gBAAiB,GAAE,CAAE,IAAG,E;EAOnC, MAAM,KAAK,gBAAiB,GAAE,CAAE,IAAG,E;EAOnC,MAAM,KAAK,gBAAiB,GACxB,MAAM,KAAK,gBAA iB,GAAE,MAAM,KAAK,gB;EA07C,MAAM,KAAK,gBAAiB,GACxB,MAAM,KAAK,gBAAiB,GAAE,C;EA0I C,MAAM,KAAK,gBAAiB,GACxB,MAAM,KAAK,gBAAiB,GAAE,MAAM,KAAK,gB;EA07C,MAAM,KAAK, gBAAiB,GACxB,MAAM,KAAK,gBAAiB,GAAE,MAAM,KAAK,gB;EA07C,MAAM,KAAK,gBAAiB,GACxB, MAAM,KAAK,gBAAiB,GAAE,C;EAIIC,MAAM,KAAK,KAAM,GAAE,MAAM,KAAK,QAAQ,CAAC,CAAD,C ;EAIIC,MAAM,KAAK,IAAK,GAAE,MAAM,KAAK,QAAQ,CAAC,CAAD,C;EAIrC,MAAM,KAAK,QAAS,GA AE,MAAM,KAAK,QAAQ,CAAC,EAAD,C;EAIzC,MAAM,KAAK,UAAW,GACIB,MAAM,KAAK,SAAS,CAA C,aAAW,GAAE,CAAd,EAAiB,UAAW,GAAE,CAA9B,C;EAIxB,MAAM,KAAK,UAAW,GAAE,MAAM,KAAK, SAAS,CAAC,CAAD,EAAI,aAAW,GAAE,CAAjB,C;EA05C,MAAM,KAAK,YAAa,GAAE,MAAM,KAAK,QAA Q,CAAC,CAAE,IAAG,EAAN,C;EAI7C,MAAM,KAAK,UAAU,MAAO,e;IAC1B,OAAO,IAAI,K;EACb,C;EAIA, MAAM,KAAK,UAAU,SAAU,e;IAC7B,OAAO,IAAI,MAAO,GAAE,MAAM,KAAK,gBAAiB,GACzC,IAAI,mB AAmB,E;EACb,C;EAGA,MAAM,KAAK,UAAU,SAAU,e;IAC7B,OAAO,IAAI,MAAO,GAAE,IAAI,K;EAC1B,

C;EAOA,MAAM,KAAK,UAAU,SAAU,wB;IAC7B,IAAI,QAAQ,SAAU,IAAG,E;IACzB,IAAI,KAAM,GAAE,CAAE,IAAG,EAAG,GAAE,KAAtB,C;MACE,MAAM,KAAK,CAAC,sBAAuB,GAAE,KAA1B,C;;;IAGb,IAAI,IAAI,OAAO,EAaf,C;MACE,OAAO,G;;;IAGT,IAAI,IAAI,WAAW,EAAhB,C;MACE,IAAI,IAAI,WAAW,CAAC,MAAM,KAAK,UAAZ,CAAnB,C;QAGE,IAAI,YAAY,MAAM,KAAK,WAAW,CAAC,KAAD,C;QACtC,IAAI,MAAM,IAAI,IAAI,CAAC,SAAD,C;QACIB,IAAI,MAAM,GAAG,SAAS,CAAC,SAAD,CAAW,SAAS,CAAC,IAAD,C;QAC1C,OAAO,GAAG,SAAS,CAAC,KAAD,CAAQ,GAAE,GAAG,MAAM,EAAE,SAAS,CAAC,KAAD,C;;;QAEjD,OAAO,GAAI,GAAE,IAAI,OAAO,EAAE,SAAS,CAAC,KAAD,C;;;IAMvC,IAAI,eAAe,MAAM,KAAK,WAAW,CAAC,IAAI,IAAI,CAAC,KAAD,EAAQ,CAAR,CAAT,C;IAEzC,IAAI,MAAM,I;IACV,IAAI,SAAS,E;IACb,OAAO,IAAP,C;MACE,IAAI,SAAS,GAAG,IAAI,CAAC,YAAD,C;MACpB,IAAI,SAAS,GAAG,SAAS,CAAC,MAAM,SAAS,CAAC,YAAD,CAAhB,CAA+B,MAAM,E;MAC9D,IAAI,SAAS,MAAM,SAAS,CAAC,KAAD,C;MAE5B,GAAI,GAAE,M;MACN,IAAI,GAAG,OAAO,EAAd,C;QACE,OAAO,MAAO,GAAE,M;;;QAEhB,OAAO,MAAM,OAAQ,GAAE,CAAvB,C;UACE,MAAO,GAAE,GAAI,GAAE,M;;;QAEjB,MAAO,GAAE,EAAG,GAAE,MAAO,GAAE,M;;;EAG7B,C;EAIA,MAAM,KAAK,UAAU,YAAa,e;IACbC,OAAO,IAAI,M;EACb,C;EAIA,MAAM,KAAK,UAAU,WAAW,EAAY,e;IAC/B,OAAO,IAAI,K;EACb,C;EAIA,MAAM,KAAK,UAAU,mBAAoB,e;IACvC,OAAQ,IAAI,KAAM,IAAG,CAAG,GACpB,IAAI,KADgB,GACR,MAAM,KAAK,gBAAiB,GAAE,IAAI,K;EACpD,C;EAOA,MAAM,KAAK,UAAU,cAAe,e;IACIC,IAAI,IAAI,WAAW,EAAhB,C;MACE,IAAI,IAAI,WAAW,CAAC,MAAM,KAAK,UAAZ,CAAnB,C;QACE,OAAO,E;;;QAEp,OAAO,IAAI,OAAO,EAAE,cAAc,E;;;MAGpC,IAAI,MAAM,IAAI,MAAO,IAAG,CAAe,GAAE,IAAI,MAAN,GAAe,IAAI,K;MAC7C,KAAK,IAAI,MAAM,EAaf,EAAMb,GAAI,GAAE,CAAzB,EAA4B,GAAG,EAA/B,C;QACE,IAAuB,CAAIB,GAAI,GAAG,CAAe,IAAG,GAAM,KAAG,CAA1B,C;UACE,K;;;MAGJ,OAAO,IAAI,MAAO,IAAG,CAAe,GAAE,GAAI,GAAE,EAAR,GAAa,GAAI,GAAE,C;;;EAE9C,C;EAIA,MAAM,KAAK,UAAU,OAAQ,e;IAC3B,OAAO,IAAI,MAAO,IAAG,CAAe,IAAG,IAAI,KAAM,IAAG,C;EACzC,C;EAIA,MAAM,KAAK,UAAU,WAAW,EAAY,e;IAC/B,OAAO,IAAI,MAAO,GAAE,C;EACtB,C;EAIA,MAAM,KAAK,UAAU,MAAO,e;IAC1B,OAAuB,CAAf,IAAI,KAAM,GAAE,CAAG,KAAG,C;EAC5B,C;EAOA,MAAM,KAAK,UAAU,WAAW,oB;IAC/B,OAAQ,IAAI,MAAO,IAAG,KAAK,MAAQ,IAAI,IAAI,KAAM,IAAG,KAAK,K;EAC3D,C;EAOA,MAAM,KAAK,UAAU,cAAe,oB;IACIC,OAAQ,IAAI,MAAO,IAAG,KAAK,MAAQ,IAAI,IAAI,KAAM,IAAG,KAAK,K;EAC3D,C;EAOA,MAAM,KAAK,UAAU,SAAU,oB;IAC7B,OAAO,IAAI,QAAQ,CAAC,KAAD,CAAQ,GAAE,C;EAC/B,C;EAOA,MAAM,KAAK,UAAU,gBAAiB,oB;IACpC,OAAO,IAAI,QAAQ,CAAC,KAAD,CAAQ,IAAG,C;EACb,C;EAOA,MAAM,KAAK,UAAU,YAAa,oB;IACbC,OAAO,IAAI,QAAQ,CAAC,KAAD,CAAQ,GAAE,C;EAC/B,C;EAOA,MAAM,KAAK,UAAU,mBAAoB,oB;IACvC,OAAO,IAAI,QAAQ,CAAC,KAAD,CAAQ,IAAG,C;EACb,C;EASA,MAAM,KAAK,UAAU,QAAS,oB;IAC5B,IAAI,IAAI,WAAW,CAAC,KAAD,CAAnB,C;MACE,OAAO,C;;;IAGT,IAAI,UAAU,IAAI,WAAW,E;IAC7B,IAAI,WAAW,KAAK,WAAW,E;IAC/B,IAAI,OAAQ,IAAG,CAAC,QAahB,C;MACE,OAAO,E;;;IAET,IAAI,CAAC,OAAQ,IAAG,QAahB,C;MACE,OAAO,C;;;IAIT,IAAI,IAAI,SAAS,CAAC,KAAD,CAAO,WAAW,EAAhC,C;MACE,OAAO,E;;;MAEP,OAAO,C;;;EAEX,C;EAIA,MAAM,KAAK,UAAU,OAAQ,e;IAC3B,IAAI,IAAI,WAAW,CAAC,MAAM,KAAK,UAAZ,CAAnB,C;MACE,OAAO,MAAM,KAAK,U;;;MAEIB,OAAO,IAAI,IAAI,EAAE,IAAI,CAAC,MAAM,KAAK,IAAZ,C;;;EAEzB,C;EAQA,MAAM,KAAK,UAAU,IAAK,oB;IAGxB,IAAI,MAAM,IAAI,MAAO,KAIE;IACzB,IAAI,MAAM,IAAI,MAAO,GAAE,K;IACvB,IAAI,MAAM,IAAI,KAAM,KAIE;IACxB,IAAI,MAAM,IAAI,KAAM,GAAE,K;IAEtB,IAAI,MAAM,KAAK,MAAO,KAIE;IAC1B,IAAI,MAAM,KAAK,MAAO,GAAE,K;IACxB,IAAI,MAAM,KAAK,KAAM,KAIE;IACzB,IAAI,MAAM,KAAK,KAAM,GAAE,K;IAEvB,IAAI,MAAM,CAAV,EAAa,MAAM,CAAnB,EAAhB,MAAM,CAA5B,EAA+B,MAAM,C;IACrC,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAIE;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAIE;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAIE;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAIE;IACf,GAAI,IAAG,K;IACP,OAAO,MAAM,KAAK,SAAS,CAAe,GAAI,IAAG,EAAl,GAAE,GAaf,EAAqB,GAAI,IAAG,EAAl,GAAE,GAAlC,C;EAC7B,C;EAQA,MAAM,KAAK,UAAU,SAAU,oB;IAC7B,OAAO,IAAI,IAAI,CAAC,KAAK,OAAO,EAAb,C;EACjB,C;EAQA,MAAM,KAAK,UAAU,SAAU,oB;IAC7B,IAAI,IAAI,OAAO,EAaf,C;MACE,OAAO,MAAM,KAAK,K;;;UACb,IAAI,KAAK,OAAO,EAAhB,C;MACL,OAAO,MAAM,KAAK,K;;;IAGpB,IAAI,IAAI,WAAW,CAAC,MAAM,KAAK,UAAZ,CAAnB,C;MACE,OAAO,KAAK,MAAM,EAAG,GAAE,MAAM,KAAK,UAAb,GAAOB,MAA

M,KAAK,K;;UACrD,IAAI,KAAK,WAAW,CAAC,MAAM,KAAK,UAAZ,CAApB,C;MACL,OAAO,IAAI,MAA
M,EAAG,GAAE,MAAM,KAAK,UAAb,GAA0B,MAAM,KAAK,K;;IAG3D,IAAI,IAAI,WAAW,EAAnB,C;MAC
E,IAAI,KAAK,WAAW,EAAPB,C;QACE,OAAO,IAAI,OAAO,EAAE,SAAS,CAAC,KAAK,OAAO,EAAb,C;;;QA
E7B,OAAO,IAAI,OAAO,EAAE,SAAS,CAAC,KAAD,CAAO,OAAO,E;;;UAExC,IAAI,KAAK,WAAW,EAAPB,
C;MACL,OAAO,IAAI,SAAS,CAAC,KAAK,OAAO,EAAb,CAAgB,OAAO,E;;IAI7C,IAAI,IAAI,SAAS,CAAC,M
AAM,KAAK,YAAZ,CAA0B,IACvC,KAAK,SAAS,CAAC,MAAM,KAAK,YAAZ,CADIB,C;MAEE,OAAO,MA
AM,KAAK,WAAW,CAAC,IAAI,SAAS,EAAG,GAAE,KAAK,SAAS,EAajC,C;;IAM/B,IAAI,MAAM,IAAI,MA
AO,KAAI,E;IACzB,IAAI,MAAM,IAAI,MAAO,GAAE,K;IACvB,IAAI,MAAM,IAAI,KAAM,KAAI,E;IACxB,IA
AI,MAAM,IAAI,KAAM,GAAE,K;IAEtB,IAAI,MAAM,KAAK,MAAO,KAAI,E;IAC1B,IAAI,MAAM,KAAK,M
AAO,GAAE,K;IACxB,IAAI,MAAM,KAAK,KAAM,KAAI,E;IACzB,IAAI,MAAM,KAAK,KAAM,GAAE,K;IAE
vB,IAAI,MAAM,CAAV,EAaA,MAAM,CAAnB,EAASB,MAAM,CAA5B,EAA+B,MAAM,C;IACrC,GAAI,IAAG
,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IAC
b,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI
,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI
,IAAG,K;IACP,GAAI,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAA
I,IAAG,GAAI,GAAE,G;IACb,GAAI,IAAG,GAAI,KAAI,E;IACf,GAAI,IAAG,K;IACP,GAAI,IAAG,GAAI,GAA
E,GAAI,GAAE,GAAI,GAAE,GAAI,GAAE,GAAI,GAAE,GAAI,GAAE,GAAI,GAAE,G;IACjD,GAAI,IAAG,K;I
ACP,OAAO,MAAM,KAAK,SAAS,CAAE,GAAI,IAAG,EAAl,GAAE,GAAf,EAaqB,GAAI,IAAG,EAAl,GAAE,
GAAIC,C;EAC7B,C;EAQA,MAAM,KAAK,UAAU,IAAK,oB;IACxB,IAAI,KAAK,OAAO,EAAb,C;MACE,MA
AM,KAAK,CAAC,kBAAD,C;;UACN,IAAI,IAAI,OAAO,EAaf,C;MACL,OAAO,MAAM,KAAK,K;;IAGpB,IAA
I,IAAI,WAAW,CAAC,MAAM,KAAK,UAAZ,CAAnB,C;MACE,IAAI,KAAK,WAAW,CAAC,MAAM,KAAK,IA
AZ,CAAKB,IACIC,KAAK,WAAW,CAAC,MAAM,KAAK,QAAZ,CADpB,C;QAEe,OAAO,MAAM,KAAK,U;;Y
ACb,IAAI,KAAK,WAAW,CAAC,MAAM,KAAK,UAAZ,CAApB,C;QACL,OAAO,MAAM,KAAK,I;;QAGIB,IA
AI,WAAW,IAAI,WAAW,CAAC,CAAD,C;QAC9B,IAAI,SAAS,QAAQ,IAAI,CAAC,KAAD,CAAO,UAAU,CAA
C,CAAD,C;QAC1C,IAAI,MAAM,WAAW,CAAC,MAAM,KAAK,KAaz,CAArB,C;UACE,OAAO,KAAK,WAA
W,EAAG,GAAE,MAAM,KAAK,IAAb,GAAoB,MAAM,KAAK,Q;;UAEzD,IAAI,MAAM,IAAI,SAAS,CAAC,K
AAK,SAAS,CAAC,MAAD,CAaf,C;UACvB,IAAI,SAAS,MAAM,IAAI,CAAC,GAAG,IAAI,CAAC,KAAD,CAA
R,C;UACvB,OAAO,M;;;UAGN,IAAI,KAAK,WAAW,CAAC,MAAM,KAAK,UAAZ,CAApB,C;MACL,OAAO,
MAAM,KAAK,K;;IAGpB,IAAI,IAAI,WAAW,EAAnB,C;MACE,IAAI,KAAK,WAAW,EAAPB,C;QACE,OAAO,
IAAI,OAAO,EAAE,IAAI,CAAC,KAAK,OAAO,EAAb,C;;QAExB,OAAO,IAAI,OAAO,EAAE,IAAI,CAAC,KA
AD,CAAO,OAAO,E;;;UAEEnC,IAAI,KAAK,WAAW,EAAPB,C;MACL,OAAO,IAAI,IAAI,CAAC,KAAK,OAAO,
EAAb,CAAgB,OAAO,E;;IAQxC,IAAI,MAAM,MAAM,KAAK,K;IACrB,IAAI,MAAM,I;IACV,OAAO,GAAG,m
BAAmB,CAAC,KAAD,CAA7B,C;MAGE,IAAI,SAAS,IAAI,IAAI,CAAC,CAAD,EAAl,IAAI,MAAM,CAAC,GA
AG,SAAS,EAAG,GAAE,KAAK,SAAS,EAAbC,CAAd,C;MAIrb,IAAI,OAAO,IAAI,KAAK,CAAC,IAAI,IAAI,C
AAC,MAAD,CAAS,GAAE,IAAI,IAAxB,C;MACpB,IAAI,QAAS,IAAK,IAAG,EAAl,GAAE,CAAF,GAAM,IAAI
,IAAI,CAAC,CAAD,EAAl,IAAK,GAAE,EAAX,C;MAIvC,IAAI,YAAY,MAAM,KAAK,WAAW,CAAC,MAAD,
C;MACtC,IAAI,YAAY,SAAS,SAAS,CAAC,KAAD,C;MACIC,OAAO,SAAS,WAAW,EAAG,IAAG,SAAS,YAA
Y,CAAC,GAAD,CAAtD,C;QACE,MAAO,IAAG,K;QACV,SAAU,GAAE,MAAM,KAAK,WAAW,CAAC,MAA
D,C;QACIC,SAAU,GAAE,SAAS,SAAS,CAAC,KAAD,C;;MAKhC,IAAI,SAAS,OAAO,EAAPB,C;QACE,SAAU,
GAAE,MAAM,KAAK,I;;MAGzB,GAAI,GAAE,GAAG,IAAI,CAAC,SAAD,C;MACb,GAAI,GAAE,GAAG,SAA
S,CAAC,SAAD,C;;IAEpB,OAAO,G;EACT,C;EAQA,MAAM,KAAK,UAAU,OAAQ,oB;IAC3B,OAAO,IAAI,SA
AS,CAAC,IAAI,IAAI,CAAC,KAAD,CAAO,SAAS,CAAC,KAAD,CAAzB,C;EActB,C;EAIA,MAAM,KAAK,UA
AU,IAAK,e;IACxB,OAAO,MAAM,KAAK,SAAS,CAAC,CAAC,IAAI,KAAAN,EAaA,CAAC,IAAI,MAAIB,C;EA
C7B,C;EAQA,MAAM,KAAK,UAAU,IAAK,oB;IACxB,OAAO,MAAM,KAAK,SAAS,CAAC,IAAI,KAAM,GAA
E,KAAK,KAAIB,EACI,IAAI,MAAO,GAAE,KAAK,MADtB,C;EAE7B,C;EAQA,MAAM,KAAK,UAAU,GAAl,o
B;IACvB,OAAO,MAAM,KAAK,SAAS,CAAC,IAAI,KAAM,GAAE,KAAK,KAAIB,EACI,IAAI,MAAO,GAAE,
KAAK,MADtB,C;EAE7B,C;EAQA,MAAM,KAAK,UAAU,IAAK,oB;IACxB,OAAO,MAAM,KAAK,SAAS,CAA
C,IAAI,KAAM,GAAE,KAAK,KAAIB,EACI,IAAI,MAAO,GAAE,KAAK,MADtB,C;EAE7B,C;EAQA,MAAM,K

AAK,UAAU,UAAW,sB;IAC9B,OAAQ,IAAG,E;IACX,IAAI,OAAQ,IAAG,CAAf,C;MACE,OAAO,I,;;MAEP,IAAI,MAAM,IAAI,K;MACd,IAAI,OAAQ,GAAE,EAAd,C;QACE,IAAI,OAAO,IAAI,M;QACf,OAAO,MAAM,KA AK,SAAS,CACvB,GAAL,IAAG,OADgB,EAeTB,IAAK,IAAG,OAAS,GAAG,GAAL,KAAG,EAAG,GAAE,OAFZ, C,;;QAI3B,OAAO,MAAM,KAAG,SAAS,CAAC,CAAD,EAAL,GAAL,IAAI,OAAQ,GAAE,EAAtB,C,;;EAGjC,C; EAQA,MAAM,KAAG,UAAU,WAAY,sB;IAC/B,OAAQ,IAAG,E;IACX,IAAI,OAAQ,IAAG,CAAf,C;MACE,OA AO,I,;;MAEP,IAAI,OAAO,IAAI,M;MACf,IAAI,OAAQ,GAAE,EAAd,C;QACE,IAAI,MAAM,IAAI,K;QACd,OA AO,MAAM,KAAG,SAAS,CACtB,GAAL,KAAL,OAAS,GAAG,IAAK,IAAI,EAAG,GAAE,OADZ,EAeVB,IAAK,I AAG,OAFc,C,;;QAI3B,OAAO,MAAM,KAAG,SAAS,CACvB,IAAK,IAAI,OAAQ,GAAE,EADI,EAeVB,IAAK,I AAG,CAAE,GAAE,CAAF,GAAM,EAFO,C,;;EAKjC,C;EAUA,MAAM,KAAG,UAAU,mBAAoB,sB;IACvC,OA AQ,IAAG,E;IACX,IAAI,OAAQ,IAAG,CAAf,C;MACE,OAAO,I,;;MAEP,IAAI,OAAO,IAAI,M;MACf,IAAI,OAA Q,GAAE,EAAd,C;QACE,IAAI,MAAM,IAAI,K;QACd,OAAO,MAAM,KAAG,SAAS,CACtB,GAAL,KAAL,OAAS ,GAAG,IAAK,IAAI,EAAG,GAAE,OADZ,EAeVB,IAAK,KAAL,OAFc,C,;;YAGtB,IAAI,OAAQ,IAAG,EAaf,C;Q ACL,OAAO,MAAM,KAAG,SAAS,CAAC,IAAD,EAAO,CAAP,C,;;QAE3B,OAAO,MAAM,KAAG,SAAS,CAA C,IAAK,KAAG,OAAQ,GAAE,EAArB,EAA0B,CAA1B,C,;;EAGjC,C;EAGA,MAAM,KAAG,UAAU,OAAQ,oB;I ACzB,OAAO,KAAM,YAAW,MAAM,KAAM,IAAG,IAAI,WAAW,CAAC,KAAD,C;EAC1D,C;EAEA,MAAM,K AAK,UAAU,gBAAiB,GAAE,MAAM,KAAG,UAAU,Q;EAE7D,MAAM,KAAG,UAAU,IAAK,e;IACtB,OAAO,I AAI,IAAI,CAAC,MAAM,KAAG,IAAZ,C;EACnB,C;EAEA,MAAM,KAAG,UAAU,IAAK,e;IACtB,OAAO,IAAI, IAAI,CAAC,MAAM,KAAG,QAAZ,C;EACnB,C;EAEA,MAAM,KAAG,UAAU,QAAS,e;IAC1B,OAAO,IAAI,SA AS,E;EACxB,C;EAEA,MAAM,KAAG,UAAU,UAAW,e;IAC5B,OAAO,I;EACX,C;EAEA,MAAM,KAAG,UAAU ,WAAY,GAAE,MAAM,KAAG,UAAU,O;EACxD,MAAM,KAAG,UAAU,IAAK,GAAE,MAAM,KAAG,UAAU,I; EAEjD,MAAM,KAAG,UAAU,QAAS,oB;IAC1B,OAAO,IAAI,MAAM,OAAO,OAAO,UAAxB,CAAmC,IAAnC, EAAyC,KAAzC,C;EACX,C;EC3zBA,MAAM,aAAc,8B;EACpB,C;EAEA,MAAM,qBAAsB,uB;IACxB,OAAO,G; EACX,C;EAEA,MAAM,aAAc,kB;IAChB,IAAI,gB;MACA,CAAE,GAAE,GAAG,E;MACP,OAAO,CAAC,MAA M,CAAC,IAAD,EAAO,SAAP,C;IACIB,C;IACA,mB;MACI,OAAO,CAAC,MAAM,CAAC,IAAD,EAAO,SAAP,C ;IACIB,C;EACJ,C;EAEA,MAAM,SAAU,mB;IACZ,yB;MACI,OAAO,OAAO,MAAO,KAALI;IAC7B,C;EACJ,C; EAEA,MAAM,aAAc,oB;IAChB,yB;MACI,OAAO,MAAM,OAAO,CAAC,MAAD,EAAS,KAAT,C;IACxB,C;EA CJ,C;EAEA,MAAM,OAAQ,iB;IACV,yB;MACI,OAAO,MAAO,IAAG,IAAK,IAAG,EAAE,CAAC,MAAD,C;IAC /B,C;EACJ,C;EAEA,MAAM,aAAc,mB;IAChB,yB;MACI,OAAO,CAAC,CAAC,MAAD,CAAS,IAAG,CAAC,CA AC,MAAD,C;IACzB,C;EACJ,C;EAEA,MAAM,qBAAsB,2C;EAC5B,C;EAEA,MAAM,YAAa,oB;IACf,OAAO,K; EACX,C;EAEA,MAAM,gBAAiB,wB;IACnB,gBAAgB,E;EACpB,C;EAEA,MAAM,oBAAqB,wB;IACvB,gBAAg B,E;EACpB,C;EAEA,MAAM,kBAAmB,wB;IACrB,gBAAgB,E;EACpB,C;EAEA,MAAM,mBAAoB,+B;IACtB,g BAAgB,E;EACpB,C,;;IAGI,MAAM,IAAI,KAAG,CACf,iDAaKd,GACID,qDAAsD,GACTd,uDAHE,C;EAIV,C;E AEA,MAAM,gBAAiB,+B;IACnB,mB;MACI,OAAO,Y;IACX,C;EACJ,C;EC/EA,MAAM,UAAW,mB;IACb,IAAI, QAAQ,OAAO,C;IACnB,IAAI,KAAM,KAAL,QAAd,C;MACI,IAAI,OAAO,CAAE,KAAL,QAajB,C;QACI,OAAO ,MAAM,gBAAgB,CAAC,CAAD,EAAL,CAAJ,C,;;MAEjC,OAAO,MAAM,mBAAmB,CAAC,CAAD,EAAL,CAAJ, C,;;IAEpC,IAAI,KAAM,KAAL,QAAS,IAAG,KAAM,KAAL,SAAP,C;MACI,OAAO,MAAM,mBAAmB,CAAC,C AAD,EAAL,CAAJ,C,;;IAEpC,OAAO,CAAC,gBAAgB,CAAC,CAAD,C;EAC5B,C;EAEA,MAAM,mBAAoB,mB;I ACtB,OAAO,CAAE,GAAE,CAAE,GAAE,EAaf,GAAO,CAAE,GAAE,CAAE,GAAE,CAAF,GAAM,C;EACpC, C;EAEA,MAAM,gBAAiB,mB;IACnB,IAAI,CAAE,GAAE,CAAR,C;MAAW,OAAO,E;IACIB,IAAI,CAAE,GAA E,CAAR,C;MAAW,OAAO,C;IAEIB,IAAI,CAAE,KAAL,CAAV,C;MACI,IAAI,CAAE,KAAL,CAAV,C;QAAa,OA AO,C;MAEpB,IAAI,KAAG,CAAE,GAAE,C;MACb,OAAO,EAAG,KAAL,CAAE,GAAE,CAAE,GAAE,CAAF,G AAO,EAAG,GAAE,CAAE,GAAE,EAaf,GAAO,C,;;IAG7C,OAAO,CAAE,KAAL,CAAE,GAAG,CAAE,KAAL,C AAE,GAAE,CAAF,GAAM,CAAJB,GAAsB,E;EACzC,C;EAEA,MAAM,QAAS,oB;IACX,OAAO,MAAM,OAAO, CAAC,KAAG,GAAC,CAAP,C;EACxB,C;EAEA,MAAM,QAAS,oB;IACX,OAAO,MAAM,OAAO,CAAC,KAAG ,GAAC,CAAP,C;EACxB,C;EAEA,MAAM,KAAM,GAAE,IAAI,KAAM,IAAG,I;EAE3B,MAAM,aAAc,GAAE,I,;; IAGIB,OAAyB,CAAhB,CAAE,GAAE,YAAY,KAAG,CAAE,GAAE,KAAP,CAAE,GAAE,CAAZ,CAAE,GAAE,K AAQ,KAAG,CAAE,GAAE,CAAP,CAAW,GAAE,C;EACxE,C;EA4DE,a;IAzDE,IAAI,MAAM,IAAI,WAAG,CAA gB,CAAhB,C;IACV,IAAI,aAAa,IAAI,YAAJ,CAAiB,GAAjB,C;IACjB,IAAI,aAAa,IAAI,YAAJ,CAAiB,GAAjB,C

;IACjB,IAAI,WAAW,IAAI,UAAJ,CAAe,GAAf,C;IACf,IAAI,WAAW,C;IACf,IAAI,YAAY,C;IAEhB,UAAU,CAAC,CAAD,CAAI,GAAE,E;IACbB,IAAI,QAAQ,CAAC,QAAD,CAAW,KAAI,CAA3B,C;MACI,QAAS,GAAE,C;MACX,SAAU,GAAE,C;;IAGhB,MAAM,aAAc,oB;MACHb,OAAO,MAAM,gBAAgB,CAAC,KAAK,CAAC,KAAD,CAAQ,GAAE,GAAF,GAAQ,KAAtB,C;IACjC,C;IAEA,MAAM,gBAAiB,oB;MACnB,UAAU,CAAC,CAAD,CAAI,GAAE,K;MACHb,OAAO,MAAM,KAAK,SAAS,CAAC,QAAQ,CAAC,QAAD,CAAT,EAAqB,QAAQ,CAAC,SAAD,CAA7B,C;IAC/B,C;IAEA,MAAM,eAAgB,oB;MACIB,QAAQ,CAAC,QAAD,CAAW,GAAE,KAAK,K;MACIB,QAAQ,CAAC,SAAD,CAAY,GAAE,KAAK,M;MAC3B,OAAO,UAAU,CAAC,CAAD,C;IACrB,C;IAEA,MAAM,YAAa,oB;MACf,OAAO,MAAM,eAAe,CAAC,KAAK,CAAC,KAAD,CAAQ,GAAE,GAAF,GAAQ,KAAtB,C;IACbC,C;IAEA,MAAM,eAAgB,oB;MACIB,UAAU,CAAC,CAAD,CAAI,GAAE,K;MACHb,OAAO,QAAQ,CAAC,CAAD,C;IACnB,C;IAEA,MAAM,cAAe,oB;MACjB,QAAQ,CAAC,CAAD,CAAI,GAAE,K;MACd,OAAO,UAAU,CAAC,CAAD,C;IACrB,C;IAGA,MAAM,cAAe,oB;MACjB,UAAU,CAAC,CAAD,CAAI,GAAE,K;MACHb,OAAO,QAAQ,CAAC,SAAD,CAAY,GAAE,a;IACjC,C;IAEA,MAAM,eAAgB,kB;MACIB,IAAc,CAAT,GAAI,GAAE,CAAG,MAAI,GAAIB,C;QACI,OAAO,GAAI,GAAE,C;;;QAGb,UAAU,CAAC,CAAD,CAAI,GAAE,G;QACHb,OAAc,CAA9B,QAAQ,CAAC,SAAD,CAAY,GAAE,EAAG,GAAE,CAAG,IAAE,QAAQ,CAAC,QAAD,CAAW,GAAE,C;;IAErE,C;EACJ,CAAe,G;EAEF,MAAM,cAAe,gB;IACjB,OAAO,CAAe,IAAG,IAAK,GAAE,CAAF,GAAM,MAAM,SAAS,E;EAC1C,C;EC9GA,IAAI,OAAO,MAAM,UAAU,WAAy,KAAI,WAA3C,C;IACI,MAAM,UAAU,WAAy,qC;MACxB,QAAS,GAAE,QAAS,IAAG,C;MACvB,OAAO,IAAI,YAAY,CAAC,YAAD,EAAe,QAAf,CAAYb,KAAI,Q;IACxD,C;;EAEJ,IAAI,OAAO,MAAM,UAAU,SAAU,KAAI,WAAzC,C;IACI,MAAM,UAAU,SAAU,qC;MACtB,IAAI,gBAAgB,IAAI,SAAS,E;MACjC,IAAI,QAAS,KAAI,SAAU,IAAG,QAAS,GAAE,aAAa,OAAtD,C;QACI,QAAS,GAAE,aAAa,O;;MAE5B,QAAS,IAAG,YAAY,O;MACxB,IAAI,YAAY,aAAa,QAAQ,CAAC,YAAD,EAAe,QAAf,C;MACrC,OAAO,SAAU,KAAI,EAAG,IAAG,SAAU,KAAI,Q;IAC7C,C;;EAGJ,IAAI,OAAO,IAAI,KAAI,MAAM,KAAI,WAAzB,C;IACI,IAAI,KAAI,MAAM,gB;MACN,CAAe,GAAE,CAAC,C;MACL,IAAI,CAAe,KAAI,CAAe,IAAG,KAAK,CAAC,CAAD,CAApB,C;QACI,OAAO,MAAM,CAAC,CAAD,C;;MAEjB,OAAO,CAAe,GAAE,CAAe,GAAE,CAAF,GAAM,E;IACvB,C;;EAEJ,IAAI,OAAO,IAAI,MAAO,KAAI,WAA1B,C;IACI,IAAI,MAAO,gB;MACP,IAAI,KAAK,CAAC,CAAD,CAAT,C;QACI,OAAO,G;;MAEX,IAAI,CAAe,GAAE,CAAR,C;QACI,OAAO,IAAI,MAAM,CAAC,CAAD,C;;MAErB,OAAO,IAAI,KAAK,CAAC,CAAD,C;IACpB,C;;EAsKF,a;IAIKE,IAAI,UAAU,qB;IACd,IAAI,iBAAiB,IAAI,KAAK,CAAC,OAAD,C;IAC9B,IAAI,iBAAiB,IAAI,KAAK,CAAC,cAAD,C;IAC9B,IAAI,uBAAuB,CAAC,GAAC,c;IAC7B,IAAI,uBAAuB,CAAC,GAAC,c;IAE7B,IAAI,OAAO,IAAI,KAAI,MAAM,KAAI,WAAzB,C;MACI,IAAI,KAAI,MAAM,gB;QACN,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAIB,C;UACI,IAAI,SAAS,C;UACb,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAIB,C;YACI,MAAO,IAAI,CAAe,GAAE,CAAe,GAAE,CAAG,GAAE,C;;UAE5B,OAAO,M;;UAEP,IAAI,IAAI,IAAI,IAAI,CAAC,CAAD,C;UACHb,IAAI,KAAK,CAAe,GAAE,C;UACb,IAAI,CAAC,QAAQ,CAAC,CAAD,CAAb,C;YAAB,OAAO,IAAI,IAAI,CAAC,CAAe,GAAE,IAAI,IAAT,C;UACjC,IAAI,CAAC,QAAQ,CAAC,EAAD,CAAb,C;YAAmB,OAAO,CAAC,IAAI,IAAI,CAAC,CAAC,CAAe,GAAE,IAAI,IAAV,C;UACnC,OAAgB,CAAR,CAAe,GAAE,EAAL,IAAE,C;;MAE1B,C;;IAEJ,IAAI,OAAO,IAAI,KAAI,MAAM,KAAI,WAAzB,C;MACI,IAAI,KAAI,MAAM,gB;QACN,IAAI,IAAI,IAAI,IAAI,CAAC,CAAD,C;QACHb,IAAI,KAAK,CAAe,GAAE,C;QACb,IAAI,CAAC,QAAQ,CAAC,CAAD,CAAI,IAAG,CAAC,QAAQ,CAAC,EAAD,CAA7B,C;UAAmC,OAAO,IAAI,IAAI,CAAC,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,IAAI,IAAnB,C;QACID,OAAgB,CAAR,CAAe,GAAE,EAAL,IAAE,C;MACtB,C;;IAGJ,IAAI,OAAO,IAAI,KAAI,MAAM,KAAI,WAAzB,C;MACI,IAAI,KAAI,MAAM,gB;QACN,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAIB,C;UACI,IAAI,SAAS,C;UACb,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAIB,C;YACI,MAAO,IAAI,CAAe,GAAE,CAAe,GAAE,CAAG,GAAE,C;;UAE5B,OAAO,M;;UAGP,IAAI,IAAI,IAAI,IAAI,CAAC,CAAC,CAAF,CAAhB,EAAsB,IAAI,IAAI,IAAI,CAAC,CAAC,CAAF,C;UACIC,OAAO,CAAe,KAAI,QAAS,GAAE,CAAF,GAAM,CAAe,KAAI,QAAS,GAAE,EAAG,GAAe,CAAP,CAAe,GAAE,CAAG,KAAI,CAAE,GAAE,CAAP,C;;MAEIE,C;;IAMJ,IAAI,OAAO,IAAI,MAAO,KAAI,WAA1B,C;MACI,IAAI,qB;QACA,IAAI,CAAe,IAAG,CAAC,cAAV,C;UAEI,IAAI,CAAe,GAAE,oBAAR,C;YAEI,IAAI,CAAe,GAAE,oBAAR,C;cAGI,OAAO,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,IAAI,I;;cAKzB,OAAO,IAAI,IAAI,CAAC,CAAe,GAAE,CAAe,GAAG,CAAE,IAAG,CAAe,GAAE,CAAP,CAAZ,C;;;YAKnB,OAAO,IAAI,IAAI,CAAC,CAAe,GAAE,IAAI,KAAK,CAAAC,CAAe,GAAE,CAAe,GAAE,CAAT,CAAd,C;;cAGIB,IAAI,CAAe,IAAG,CAAC,cAAV,C;UAED,OAAO,CAA

C,KA AK,CAAC,CAAC,CAAF,C;;;UAKb,IAAI,SAAS,C;UACb,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,IAAG,cA
AnB,C;YAEI,IAAI,KA AK,CAAE,GAAE,CAAE,GAAE,C;YAEjB,MAAO,IAAG,EAAG,GAAE,C;;UAEnB,OAA
O,M;;MAEf,C;MACA,IAAI,MAAO,GAAE,K;;IAEjB,IAAI,OAAO,IAAI,MAAO,KA AI,WAA1B,C;MACI,IAAI,
MAAO,gB;QACP,IAAI,CAAE,GAAE,CAAR,C;UAEI,OAAO,G;;cAEN,IAAI,CAAE,GAAE,CAAE,IAAG,cAAb,
C;UAED,IAAI,CAAE,GAAE,oBAAR,C;YAGI,OAAO,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,IAAI,I;;;YAIzB,O
AAO,IAAI,IAAI,CAAC,CAAE,GAAE,IAAI,KA AK,CAAC,CAAE,GAAE,CAAE,GAAE,CAAT,CAAd,C;;;UAK
nB,IAAI,IAAI,IAAI,KA AK,CAAC,CAAE,GAAE,CAAL,C;UAEjB,IAAI,SAAS,C;UACb,IAAI,CAAE,IAAG,cAA
T,C;YAEI,IAAI,KA AK,CAAE,GAAE,CAAE,GAAE,C;YAEjB,MAAO,IAAG,EAAG,GAAE,E;;UAGnB,OAAO,I
AAI,KA AK,CAAC,CAAD,CAAI,GAAE,M;;MAE9B,C;;IAEJ,IAAI,OAAO,IAAI,MAAO,KA AI,WAA1B,C;MAC
I,IAAI,MAAO,gB;QACP,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAIB,C;UACI,IAAI,SAAS,C;UACb,IA
AI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAIB,C;YACI,MAAO,IAAI,CAAE,GAAE,CAAE,GAAE,CAAG,GA
AE,C;;UA E5B,OAAO,M;;QAEX,OAAO,IAAI,IAAI,CAAS,CAAP,CAAE,GAAE,CAAG,KAAG,CAAE,GAAE,C
AAP,CAAT,CAAoB,GAAE,C;MACzC,C;;IAEJ,IAAI,OAAO,IAAI,MAAO,KA AI,WAA1B,C;MACI,IAAI,MAA
O,gB;QACP,IAAI,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAIB,C;UACI,IAAI,KA AK,CAAE,GAAE,C;UACb,I
AAI,KA AK,EAAG,GAAE,C;UACd,IAAI,KA AK,EAAG,GAAE,C;UAEd,OAAQ,CAAC,EAAG,GAAE,CAAE,G
AAE,EAAG,GAAE,CAAE,GAAE,EAAG,GAAE,CAAE,GAAE,C;;QAExC,OAAO,IAAI,IAAI,CAAC,CAAE,GA
AE,CAAL,C;MACnB,C;;IAEJ,IAAI,OAAO,IAAI,MAAO,KA AI,WAA1B,C;MACI,IAAI,MAAO,gB;QACP,IAAI,
IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,cAAIB,C;UACI,IAAI,KA AK,CAAE,GAAE,C;UACb,IAAI,KA AK,EA
G,GAAE,C;UACd,IAAI,KA AK,EAAG,GAAE,C;UAEd,OAAQ,EAAG,GAAE,EAAG,GAAE,EAAG,GAAE,CAA
E,GAAE,EAAG,GAAE,CAAE,GAAE,C;;QAExC,OAAO,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,C;MACzB,C;;E
AER,CAAE,G;EACF,IAAI,OAAO,IAAI,MAAO,KA AI,WAA1B,C;IACI,IAAI,MAAO,e;MACP,IAAI,IAAI,C;MA
CR,IAAI,SAAS,SAAS,O;MAEtB,KA AK,IAAI,IAAI,CAAb,EAAGb,CAAE,GAAE,MAApB,EA A4B,CAAC,EA
7B,C;QACI,IAAI,SAAS,CAAC,CAAD,CAAI,KA AI,QAAS,IAAG,SAAS,CAAC,CAAD,CAAI,KA AI,CAAC,QA
AnD,C;UACI,OAAO,Q;;QAEX,CAAE,IAAG,SAAS,CAAC,CAAD,CAAI,GAAE,SAAS,CAAC,CAAD,C;;MAEj
C,OAAO,IAAI,KA AK,CAAC,CAAD,C;IACpB,C;;EAEJ,IAAI,OAAO,IAAI,MAAO,KA AI,WAA1B,C;IACI,IAAI
,MAAO,gB;MACP,OAAO,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,IAAI,O;IAC7B,C;;EAEJ,IAAI,OAAO,IAAI,K
AAM,KA AI,WAAzB,C;IACI,IAAI,KAAM,gB;MACN,OAAO,IAAI,IAAI,CAAC,CAAD,CAAI,GAAE,IAAI,M;I
AC7B,C;;EAEJ,IAAI,OAAO,IAAI,MAAO,KA AI,WAA1B,C;IACI,IAAI,MAAO,GAQT,oB;MAPE,oB;QACI,IAA
I,SAAS,CAAE,KA AI,C;QACnB,IAAI,MAAO,KA AI,CAAf,C;UACI,OAAO,E;;QAEX,OAAO,EAAG,IAAG,GAA
G,CAAC,MAAD,CAAS,GAAE,GA AI,GAAE,CAA vB,CAA0B,GAAE,C;MAC1C,C;IACJ,CAAE,CAAC,IAAI,IA
AL,EA AW,IAAI,IAAf,C;;EAIN,IAAI,OAAO,WAAW,OAAQ,KA AI,WAAIC,C;IACI,WAAW,OAAQ,gB;MACf,
OAAO,CAAE,IAAG,IAAK,IAAG,CAAC,UAAW,IAAG,IAAK,IAAG,CAAC,UAAU,UAAW,KA AI,SAAS,UAA
U,U;IAC5F,C;;EAGJ,IAAI,OAAO,KA AK,UAAU,KAAM,KA AI,WAApC,C;IAEI,KA AK,UAAU,KAAM,e;MAEn
B,IAAI,IAAK,IAAG,IAAZ,C;QACE,MAAM,IAAI,SAAJ,CAAc,6BAAd,C;;MAGR,IAAI,IAAI,MAAM,CAAC,IA
AD,C;MAGd,IAAI,MAAM,CAAC,OAAQ,KA AI,C;MAGvB,IAAI,QAAQ,SAAS,CAAC,CAAD,C;MACrB,IAAI,
gBAAgB,KAAM,IAAG,C;MAG7B,IAAI,IAAI,aAAc,GAAE,CAAE,GACxB,IAAI,IAAI,CAAC,GA AI,GAAE,aA
AP,EAAsB,CAAtB,CADgB,GAExB,IAAI,IAAI,CAAC,aAAD,EAAGb,GA AhB,C;MAGV,IAAI,MAAM,SAAS,C
AAC,CAAD,C;MACnB,IAAI,cAAc,GA AI,KA AI,SAAU,GACIC,GADkC,GAC5B,GA AI,IAAG,C;MAGf,IAAI,Q
AAQ,WAA Y,GAAE,CAAE,GAC1B,IAAI,IAAI,CAAC,GA AI,GAAE,WAAP,EA AoB,CAApB,CADkB,GA E1B,I
AAI,IAAI,CAAC,WAAD,EA Ac,GAAd,C;MAGV,OAAO,CAAE,GAAE,KA AX,C;QACE,CAAC,CAAC,CAAD,C
AAI,GAAE,K;QACP,CAAC,E;;MAIH,OAAO,C;IAC T,C;;EAIFF,a;;MA5EM,IAAI,MAAO,GAAE,CAAb,C;QAA
gB,OAAO,IAAI,IAAI,CAAC,CAAD,EA AI,MAAO,GAAE,MAAb,C;MAC/B,OAAO,IAAI,IAAI,CAAC,MAAD,E
AAS,MAAT,C;IACnB,C;;MAEI,IAAI,OAAO,GA AI,KA AI,WAA nB,C;QACI,GA AI,GAAE,IAAI,O;;MAEd,KAA
M,GAAE,eAAe,CAAC,KAAM,IAAG,CAAV,EA Aa,IAAI,OAAjB,C;MACvB,GA AI,GAAE,IAAI,IAAI,CAAC,K
AAD,EA AQ,eAAe,CAAC,GAAD,EAAM,IAAI,OAAV,CAA vB,C;MACd,OAAO,IAAI,IAAI,YAAR,CAAqB,IAA
I,SAAS,CAAC,KAAD,EA AQ,GAAR,CAAIC,C;IACX,C;IAEA,IAAI,SAAS,CAAC,SAAD,EAAY,UAAZ,EA Aw
B,WAAxB,EA AqC,UAArC,EA AiD,YAAjD,EA A+D,YAA/D,C;IACb,KA AK,IAAI,IAAI,CAAb,EA AgB,CAAE,G
AAE,MAAM,OAA1B,EA AmC,EA AE,CAArC,C;MACI,IAAI,aAAa,MAAM,CAAC,CAAD,C;MACvB,IAAI,OA

AO,UAAU,UAAU,KAAM,KAAl,WAAzC,C;QACI,UAAU,UAAU,KAAM,GAAE,KAAK,UAAU,K;;MAE/C,IAA
I,OAAO,UAAU,UAAU,MAAO,KAAl,WAA1C,C;QACI,MAAM,eAAe,CAAC,UAAU,UAAU,EAAB,OAAB,E
AAgC,QAC1C,eAD0C,CAAhC,C;;;MAQJ,CAAN,Y;MAAF,CAAE,MAAM,CAAC,IAAD,EAAO,IAAl,UAAJ,C
AAe,CAAf,CAAP,E;;;MAErB,IAAl,QAAQ,QAAQ,UAAU,M;MAC9B,MAAM,eAAe,CAAC,QAAQ,UAAT,EA
qB,OAARb,EAAB,+B;QAE3C,OAAO,KAAK,KAAK,CAAC,IAAD,EAAO,IAAP,EAaA,EAAB,MAAM,KAAK,
CAAC,KAAD,CAAlB,C;MACrB,CAH+C,CAA9B,C;;IASzB,KAAK,IAAl,IAAl,CAAb,EAAB,CAAE,GAAE,M
AAM,OAA1B,EAAMC,EAAB,CAArC,C;MACI,IAAl,aAAa,MAAM,CAAC,CAAD,C;MACvB,IAAl,OAAO,UA
AU,UAAU,IAAK,KAAl,WAAxC,C;QACI,MAAM,eAAe,CAAC,UAAU,UAAU,EAAB,KAAB,EAAB,kC;UA
E3C,OAAO,EAAB,MAAM,KAAK,CAAC,IAAD,CAAM,IAAl,CAAC,QAAD,EAAB,IAAX,C;QAC1C,CAH+C,C
AA9B,C;;;IAU7B,IAAl,uC;MACA,IAAl,CAAE,GAAE,CAAR,C;QAAW,OAAO,E;MACIB,IAAl,CAAE,GAAE,
CAAR,C;QAAW,OAAO,C;MAE1B,IAAl,CAAE,KAAl,CAAV,C;QACI,IAAl,CAAE,KAAl,CAAV,C;UAAa,OA
O,C;QAEpB,IAAl,KAAK,CAAE,GAAE,C;QACb,OAAO,EAAG,KAAl,CAAE,GAAE,CAAE,GAAE,CAAF,GAA
O,EAAG,GAAE,CAAE,GAAE,EAAB,GAAO,C;;MAG7C,OAAO,CAAE,KAAl,CAAE,GAAG,CAAE,KAAl,CAA
E,GAAE,CAAF,GAAM,CAAjB,GAAsB,E;IACzC,C;IAEA,KAAK,IAAl,IAAl,CAAb,EAAB,CAAE,GAAE,MA
AM,OAA1B,EAAMC,EAAB,CAArC,C;MACI,IAAl,aAAa,MAAM,CAAC,CAAD,C;MACvB,IAAl,OAAO,UA
AU,UAAU,KAAM,KAAl,WAAzC,C;QACI,MAAM,eAAe,CAAC,UAAU,UAAU,EAAB,MAAB,EAAB,mC;U
AE5C,OAAO,KAAK,UAAU,KAAK,KAAK,CAAC,IAAD,EAAO,eAAB,IAAG,oBAA1B,C;QACpC,CAHgD,CA
A/B,C;;EAOjC,CAAE,G;EC/WF,MAAM,KAAM,GAAE,QACH,OADG,aAEC,WAFD,UAGF,QAHE,C;EAMd,M
AAM,WAAy,8C;IACd,IAAl,qBAAB,MAAM,yBAAB,CAAC,KAAD,EAAB,YAAR,C;IACxD,IAAl,kBAAB
,IAAG,IAAK,IAAG,kBAAB,IAAK,IAAG,IAA5D,C;MACI,OAAO,kBAAB,IAAl,KAAK,CAAC,UAAD,C;;IA
GtC,kBAAB,GAAE,MAAM,yBAAB,CAAC,UAAD,EAAB,YAAB,C;IACpD,IAAl,kBAAB,IAAG,IAAK,IAA
G,OAAQ,IAAG,kBAAB7C,C;MACI,OAAO,UAAU,CAAC,YAAD,C;;IAGrB,OAAO,MAAM,WAAW,CAAC,UA
AD,EAAB,MAAM,eAAe,CAAC,KAAD,CAAlC,EAAB2C,YAA3C,C;EAC5B,C;EAEA,MAAM,WAAy,qD;IACd,IA
Al,qBAAB,MAAM,yBAAB,CAAC,KAAD,EAAB,YAAR,C;IACxD,IAAl,kBAAB,IAAG,IAAK,IAAG,kBA
AB,IAAK,IAAG,IAA5D,C;MACI,kBAAB,IAAl,KAAK,CAAC,UAAD,EAAB,KAAB,C;MAC3B,M;;IAGJ,kBA
AB,GAAE,MAAM,yBAAB,CAAC,UAAD,EAAB,YAAB,C;IACpD,IAAl,kBAAB,IAAG,IAAK,IAAG,OAAQ,I
AAG,kBAAB7C,C;MACI,UAAU,CAAC,YAAD,CAAE,GAAE,K;MAC3B,M;;IAGJ,MAAM,WAAW,CAAC,UA
AD,EAAB,MAAM,eAAe,CAAC,KAAD,CAAlC,EAAB2C,YAA3C,EAABD,KAABD,C;EACrB,C;;IAGI,IAAl,IAAK,
KAAl,KAAB,C;MAAOB,OAAO,I;IAE3B,IAAl,WAAW,IAAl,W;IACnB,IAAl,QAAS,IAAG,IAAhB,C;MACI,IAA
I,aAAa,QAAQ,W;MACzB,KAAK,IAAl,IAAl,CAAb,EAAB,CAAE,GAAE,UAAU,OAA9B,EAABuC,CAAC,EA
xC,C;QACI,IAAl,OBAA0B,CAAC,UAAU,CAAC,CAAD,CAAX,EAAB,KAAB,CAA9B,C;UACI,OAAO,I;;;IA
KnB,IAAl,iBAAB,IAAl,UAAW,IAAG,IAAK,GAAE,MAAM,eAAe,CAAC,IAAl,UAAU,CAAAB,GAA0C,I;IACt
F,IAAl,mBAAB,cAAe,IAAG,IAAK,GAAE,cAAc,YAAhB,GAA+B,I;IAC7E,OAAO,gBAAB,IAAG,IAAK,IAA
G,OBAA0B,CAAC,gBAAD,EAAB,KAAB,C;EACjE,C;EAQA,MAAM,OAAQ,4B;IACV,IAAl,KAAM,KAAl,
MAAD,C;MACI,QAAQ,OAAO,MAAf,C;aACS,Q;aACA,Q;aACA,S;aACA,U;UACD,OAAO,I;gBAEP,OAAO,M
AAO,YAAW,M;;;IAIrC,IAAl,MAAO,IAAG,IAAK,IAAG,KAAM,IAAG,IAAK,KAAl,OAAO,MAAO,KAAl,QA
AS,IAAG,OAAO,MAAO,KAAl,UAApD,CAApC,C;MACI,OAAO,K;;IAGX,IAAl,OAAO,KAAM,KAAl,UAAW,
IAAG,MAAO,YAAW,KAARd,C;MACI,OAAO,I;;IAGX,IAAl,QAAQ,MAAM,eAAe,CAAC,KAAD,C;IACjC,IA
Al,cAAc,KAAM,IAAG,IAAK,GAAE,KAAK,YAAP,GAAsB,I;IACtD,IAAl,WAAy,IAAG,IAAK,IAAG,YAAa,IA
AG,WAA3C,C;MACI,IAAl,WAAW,WAAW,W;MAC1B,IAAl,QAAQ,KAAM,KAAl,MAAM,KAAK,OAAjC,C;
QACI,OAAO,MAAO,KAAl,K;;IAI1B,IAAl,gBAAB,KAAK,W;IAGzB,IAAl,aAAc,IAAG,IAARb,C;MACI,OA
AO,MAAO,YAAW,K;;IAG7B,IAAl,aAAa,KAAM,KAAl,MAAM,KAAK,UAAW,IAAG,MAAM,YAAa,IAAG,IA
A1E,C;MACI,OAAO,OBAA0B,CAAC,MAAM,YAAP,EAAB,KAARb,C;;IAGrC,OAAO,K;EACX,C;EAEA,MA
AM,SAAU,gB;IACZ,OAAO,OAAO,CAAE,IAAG,QAAS,IAAG,CAAE,YAAW,MAAM,K;EACtD,C;EAEA,MA
AM,OAAQ,oB;IACV,OAAO,KAAM,YAAW,MAAM,U;EACIC,C;EAEA,MAAM,aAAc,oB;IACHb,IAAl,OAAO,
OAAO,K;IAE1B,OAAO,IAAK,KAAl,QAAS,IACIB,IAAK,KAAl,SAAU,IACnB,MAAM,SAAS,CAAC,KAAD,C
AAQ,IACvB,MAAM,OAAO,CAAC,KAAD,EAAB,MAAM,OAAO,WAArB,C;EACxB,C;EAEA,MAAM,eAAB,
oB;IACIB,OAAO,OAAO,KAAM,KAAl,QAAS,IAAG,MAAM,OAAO,CAAC,KAAD,EAAB,MAAM,OAAO,aAA

rB,C;EACrD,C;,,,,,;aCpD2C,gB;,,,,;MCnDvC,4B;MAjBA,AA6C,E;MAC7C,gBAAgD,C;IAFpD,C;;MAKgB,iB;I
AAA,C;;MAGA,oB;IAAA,C;;MAE6B,OAAQ,0BAAR,YAAQ,EAAU,KAAM,QAAbB,C;IAAR,C;;MAEN,gBAA
S,K;IAAT,C;;MAEJ,OAAnc,MAAmC,kBAA8B,IAA9B,C;IAAA,C;;MAEL,gB;IAAA,C;;MAE1B,8B;IAAA,C;,,,;
MAAA,qC;QAAA,oB;;MAAA,8B;IAAA,C;,,,;MDdwC,oBAAa,MAAS,IAAT,CAAb,EAA6B,SAA7B,C;IAAA,C;g
EAE5C,yB;MAAA,mB;MAAA,6B;QAC2D,YAAa,QAAS,IAAT,C;QAIvD,Q;QAAA,OAAA,KAAM,OAAN,GAA
a,CAAb,I;QAAb,aAAU,CAAV,iB;UACI,MAAM,CAAN,IALgF,IAKrE,CAAK,CAAL,C;;QALwC,OAOhD,K;MA
PgD,C;KAD3D,C;;MAKiB,Q;MAAA,OAAA,KAAM,OAAN,GAAa,CAAb,I;MAAb,aAAU,CAAV,iB;QACI,MA
AM,CAAN,IAAW,KAAC,CAAL,C;;MAEf,OAAO,K;IACX,C;;MAMiB,IAAN,I;MAFP,aAAsB,MAAe,IAAf,C;M
ACtB,gBAAkB,c;MAEd,IADS,IACT,mBADs,IACT,EAAM,IAAN,E;QAAc,oBAAa,MAAb,EAAqB,KAAR,C;W
ACd,WAFS,IAET,S;QAAS,a;;QAZA,U;QAAA,SAaqB,MABf,OAAN,GAAa,CAAb,I;QAAb,aAAU,CAAV,mB;U
AakC,MAZ9B,CAAM,CAAN,IAYS,C,IAZ3B,CAAK,CAAL,C;;QAYH,OAAsB,M;;MAHIC,W;IAKJ,C;2EAEA,yB
;MAAA,iC;MAAA,6B;QACoF,YAAa,aAAa,IAAb,EAAM,B,KAAN,B,C;QAIbHf,Q;QAAA,OAAA,KAAM,OAAN,
GAAa,CAAb,I;QAAb,aAAU,CAAV,iB;UACI,MAAM,CAAN,IAiBoH,IAjBzG,CAAK,CAAL,C;;QAIbIE,OAfzE,
K;MAeyE,C;KADPf,C;;MAQiB,IAAN,I;MAFP,aAAa,IAAb,WAAa,CAAD,IAAC,C;MACb,gBAAkB,W;MAEd,I
ADS,IACT,mBADs,IACT,EAAM,IAAN,YADS,IACT,EAAY,KAaz,E;QAAqB,a;;QA1BZ,U;QAAA,SA2BkB,M
A3BZ,OAAN,GAAa,CAAb,I;QAAb,aAAU,CAAV,mB;UA2B+B,MA1B3B,CAAM,CAAN,IA0BmC,IA1BxB,CA
AK,CAAL,C;;QA0BH,OAAM,B,M;;MAF/B,W;IAIJ,C;qEAEA,yB;MAAA,2B;MAAA,gC;MAAA,6B;QAGiB,Q;Q
ADb,YAAY,UAAU,IAAV,EAAGB,IAAhB,C;QACC,OAAA,KAAM,OAAN,GAAa,CAAb,I;QAAb,aAAU,CAAV,
iB;UACI,YACY,eAAK,CAAL,E;UACpB,KAAC,CAAD,CAAG,GAAG,K;;QAEP,OAAO,K;MACX,C;KA
TA,C;mFAWA,yB;MAAA,mB;MAAA,gC;MAAA,6B;QAGiB,Q;QADb,YAAY,QAAY,IAAZ,C;QACC,OAAA,K
AAM,OAAN,GAAa,CAAb,I;QAAb,aAAU,CAAV,iB;UACI,YACY,eAAK,CAAL,E;UACpB,KAAC,CAA
D,CAAG,GAAG,K;;QAEP,OAAO,K;MACX,C;KATA,C;;MAeiB,IAAN,I;MAFP,aAAsB,MAAY,IAAZ,C;MACt
B,gBAAkB,W;MAEd,IADS,IACT,mBADs,IACT,EAAM,IAAN,E;QAAc,oBAAa,MAAb,K;WACd,WAFS,IAET,
S;QAAS,a;;QA3DA,U;QAAA,SA4DkB,MA5DZ,OAAN,GAAa,CAAb,I;QAAb,aAAU,CAAV,mB;UA4D+B,MA3
D3B,CAAM,CAAN,IA2DmC,IA3DxB,CAAK,CAAL,C;;QA2DH,OAAM,B,M;;MAH/B,W;IAKJ,C;qEAEA,yB;M
AAA,2B;MAAA,6B;QAC2E,YAAa,UAAU,IAAV,EAAGB,KAAbB,C;QAJEvE,Q;QAAA,OAAA,KAAM,OAAN,
GAAa,CAAb,I;QAAb,aAAU,CAAV,iB;UACI,MAAM,CAAN,IAgEwG,IAhE7F,CAAK,CAAL,C;;QAAGwD,OA9
DhE,K;MA8DgE,C;KAD3E,C;;MAIiB,Q;MAAA,OAAA,KAAM,OAAN,GAAa,CAAb,I;MAAb,aAAU,CAAV,iB;
QACI,MAAM,CAAN,IAAW,S;;MAEf,OAAO,K;IACX,C;;MEnFA,qC;MAEI,iBACJ,MAA4B,U;MAExB,iBACJ,
MAA4B,U;MAExB,yBACJ,MAAoC,kB;MAEhC,yBACJ,MAAoC,kB;MAEhC,WACJ,MAAsB,I;IAftB,C;,,,;MAA
A,4C;QAAA,2B;;MAAA,qC;IAAA,C;;MAkBA,oC;MAEI,iBACJ,MAA2B,U;MAEvB,iBACJ,MAA2B,U;MAEvB,
yBACJ,MAAmC,kB;MAE/B,yBACJ,MAAmC,kB;MAE/B,WACJ,MAAqB,I;IAfrB,C;,,,;MAAA,2C;QAAA,0B;;M
AAA,oC;IAAA,C;;MAkBA,kC;MAEI,iBACqB,W;MAErB,iBACqB,U;MAErB,kBACuB,C;MAEvB,iBACsB,E;IA
Z1B,C;,,,;MAAA,yC;QAAA,wB;;MAAA,kC;IAAA,C;;MAeA,mC;MAEI,iBACJ,MAAM,KAAoB,U;MAEtB,iBAC
J,MAAM,KAAoB,U;MAEtB,kBACuB,C;MAEvB,iBACsB,E;IAZ1B,C;,,,;MAAA,0C;QAAA,yB;;MAAA,mC;IAA
A,C;;MAeA,oC;MAEI,iBACuB,U;MAEvB,iBACuB,K;MAEvB,kBACuB,C;MAEvB,iBACsB,E;IAZ1B,C;,,,;MAA
A,2C;QAAA,0B;;MAAA,oC;IAAA,C;;MAeA,mC;MAEI,iBACsB,Q;MAEtB,iBACsB,G;MAEtB,kBACuB,C;MA
EvB,iBACsB,C;IAZ1B,C;,,,;MAAA,0C;QAAA,yB;;MAAA,mC;IAAA,C;;MAeA,mC;MAEI,iBACmC,C;MAEnC,i
BACmC,K;MAEnC,0BAC4C,K;MAE5C,0BAC4C,K;MAE5C,yBAC2C,K;MAE3C,yBAC2C,K;MAE3C,qBACuC
,uB;MAEvC,qBACuC,sB;MAEvC,kBACuB,C;MAEvB,iBACsB,E;IA9B1B,C;,,,;MAAA,0C;QAAA,yB;;MAAA,m
C;IAAA,C;;MAiCA,qC;IAAA,C;,,,;MAAA,4C;QAAA,2B;;MAAA,qC;IAAA,C;;MAEA,sC;IAAA,C;,,,;MAAA,6C;
QAAA,4B;;MAAA,sC;IAAA,C;,,,;aC0hcoB,gB;,,,;cCpb0C,mB;;gBAyEvC,yB;eAAYB,wB;
;uBAgBzB,gC;sBAAwB,+B;mCA4JjC,qB;mCA5ImC,qB;;kBAQ1B,2B;iBAA0B,0B;,,,;eCxZgB,wB;sBCQA,sB;i
BCPA,0B;,,,;aC1K8B,e;,,,;gCC1ChD,yC;+BCTA,uC;+BCDA,sC;;gCC+I/B,+B;gBCIEqE,yB;,,,;
;,,,;
;,,,;MC1EE,wC;IANvF,C;IAAA,wD;MAO2C,8B;IAAA,C;IAP3C,8E;;MCSuF,wC;IARvF,C;IAAA,w
D;MAS2C,8B;IAAA,C;IAT3C,8E;;MZcI,OAAO,UAAI,CAAJ,C;IACX,C;;MAOI,OAAO,UAAI,CAAJ,C;IACX,C;
;MAOI,OAAO,UAAI,CAAJ,C;IACX,C;;MAOI,OAAO,UAAI,CAAJ,C;IACX,C;;MAOI,OAAO,UAAI,CAAJ,C;IA

O,CAAU,OAAV,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;MA1XP,yB;IACJ,C;;MAOW,sB;;
QAYXS,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAzXH,SAyXO,CAAU,OAAV,CAAJ,C;YAAwB,qBAAO,O;Y
AAP,uB;;;QAC9C,qBAAO,I;;;MA1XP,yB;IACJ,C;;MAOW,sB;;QAYXS,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,
IAzXH,SAyXO,CAAU,OAAV,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;MA1XP,yB;IACJ,C;
gFAEA,yB;MA0XA,oC;MAAA,gC;MA1XA,uC;QAKW,sB;;UAYXS,Q;UAAhB,iD;YAAgB,cAAhB,0B;YAAsB,I
AzXH,SAyXO,CAAU,oBAAV,CAAJ,C;cAAwB,qBAAO,O;cAAP,uB;;;UAC9C,qBAAO,I;;;QAIXP,yB;MACJ,C;
KANA,C;sFAQA,yB;MAssCA,0D;MAAA,+C;MAtsCA,uC;QAKW,qB;;UAqsCO,Q;UAAA,OAAa,SAAR,sBAAQ
,CAAb,W;UAAAd,OAAc,cAAAd,C;YAAc,uB;YACV,cAAc,UAAK,KAAL,C;YACd,IAvsCc,SAusCV,CAAU,OAAV
,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;;QAZsCP,wB;MACJ,C;KANA,C;wFAQA,yB;MAysC
A,0D;MAAA,+C;MAzsCA,uC;QAKW,qB;;UAwsCO,Q;UAAA,OAAa,SAAR,sBAAQ,CAAb,W;UAAAd,OAAc,cA
Ad,C;YAAc,uB;YACV,cAAc,UAAK,KAAL,C;YACd,IA1sCc,SA0sCV,CAAU,OAAV,CAAJ,C;cAAwB,oBAAO,
O;cAAP,sB;;;UAE5B,oBAAO,I;;;QA5sCP,wB;MACJ,C;KANA,C;wFAQA,yB;MA4sCA,0D;MAAA,+C;MA5sCA
,uC;QAKW,qB;;UA2sCO,Q;UAAA,OAAa,SAAR,sBAAQ,CAAb,W;UAAAd,OAAc,cAAAd,C;YAAc,uB;YACV,cA
Ac,UAAK,KAAL,C;YACd,IA7sCc,SA6sCV,CAAU,OAAV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBA
AO,I;;;QA/sCP,wB;MACJ,C;KANA,C;wFAQA,yB;MA+sCA,0D;MAAA,+C;MA/sCA,uC;QAKW,qB;;UA8sCO,
Q;UAAA,OAAa,SAAR,sBAAQ,CAAb,W;UAAAd,OAAc,cAAAd,C;YAAc,uB;YACV,cAAc,UAAK,KAAL,C;YACd,
IAhtCc,SAgtCV,CAAU,OAAV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;;QAltCP,wB;MACJ,C;
KANA,C;wFAQA,yB;MAktCA,0D;MAAA,+C;MAltCA,uC;QAKW,qB;;UAitCO,Q;UAAA,OAAa,SAAR,sBAAQ
,CAAb,W;UAAAd,OAAc,cAAAd,C;YAAc,uB;YACV,cAAc,UAAK,KAAL,C;YACd,IAntCc,SAmtCV,CAAU,OAAV
,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;;QArtCP,wB;MACJ,C;KANA,C;wFAQA,yB;MAqtC
A,0D;MAAA,+C;MArtCA,uC;QAKW,qB;;UAotCO,Q;UAAA,OAAa,SAAR,sBAAQ,CAAb,W;UAAAd,OAAc,cAA
d,C;YAAc,uB;YACV,cAAc,UAAK,KAAL,C;YACd,IAttCc,SAstCV,CAAU,OAAV,CAAJ,C;cAAwB,oBAAO,O;c
AAP,sB;;;UAE5B,oBAAO,I;;;QAXtCP,wB;MACJ,C;KANA,C;wFAQA,yB;MAwtCA,0D;MAAA,+C;MAXtCA,uC
;QAKW,qB;;UAutCO,Q;UAAA,OAAa,SAAR,sBAAQ,CAAb,W;UAAAd,OAAc,cAAAd,C;YAAc,uB;YACV,cAAc,U
AAK,KAAL,C;YACd,IAztCc,SAytCV,CAAU,OAAV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;
;QA3tCP,wB;MACJ,C;KANA,C;wFAQA,yB;MA2tCA,0D;MAAA,+C;MA3tCA,uC;QAKW,qB;;UA0tCO,Q;UAA
A,OAAa,SAAR,sBAAQ,CAAb,W;UAAAd,OAAc,cAAAd,C;YAAc,uB;YACV,cAAc,UAAK,KAAL,C;YACd,IA5tCc,
SA4tCV,CAAU,OAAV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;;QA9tCP,wB;MACJ,C;KANA
,C;wFAQA,yB;MA8tCA,0D;MAAA,+C;MAAA,oC;MA9tCA,uC;QAKW,qB;;UA6tCO,Q;UAAA,OAAa,SAAR,s
BAAQ,CAAb,W;UAAAd,OAAc,cAAAd,C;YAAc,uB;YACV,cAAc,UAAK,KAAL,C;YACd,IA/tCc,SA+tCV,CAAU,o
BAAV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;;QajuCP,wB;MACJ,C;KANA,C;;MAaI,IAyiL
O,qBAAQ,CAziLf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;IACX,C;;MAOI,IAqiL
O,qBAAQ,CAviLf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;IACX,C;;MAOI,IAqiL
O,qBAAQ,CAriLf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;IACX,C;;MAOI,IAiiL
O,qBAAQ,CAjiLf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;IACX,C;;MAOI,IA+hL
O,qBAAQ,CA/hLf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;IACX,C;;MAOI,IA6hL
O,qBAAQ,CA7hLf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;IACX,C;;MAOI,IA2h
LO,qBAAQ,CA3hLf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;IACX,C;;MAOI,IAy
hLO,qBAAQ,CAzhLf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAK,CAAL,C;IACX,C;kFAEA,yB
;MAAA,iE;MAAA,uC;QAKoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UAAsB,IAAI,UAU,
OAAV,CAAJ,C;YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAPA,C;kFASA,yB;MAA
A,iE;MAAA,uC;QAKoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UAAsB,IAAI,UAU,OAA
V,CAAJ,C;YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAPA,C;mFASA,yB;MAAA,iE;
MAAA,uC;QAKoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UAAsB,IAAI,UAU,OAAV,CA
AJ,C;YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAPA,C;mFASA,yB;MAAA,iE;MAA
A,uC;QAKoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UAAsB,IAAI,UAU,OAAV,CAAJ,C;
YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAPA,C;mFASA,yB;MAAA,iE;MAAA,uC

;QAKoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UAAsB,IAAI,UAAU,OAAV,CAAJ,C;YAA
wB,OAAO,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAPA,C;mFASA,yB;MAAA,iE;MAAA,uC;QA
KoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UAAsB,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,
OAAO,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAPA,C;mFASA,yB;MAAA,iE;MAAA,uC;QAKoB,
Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UAAsB,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAA
O,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAPA,C;mFASA,yB;MAAA,iE;MAAA,uC;QAKoB,Q;Q
AAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UAAsB,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;
;QACrD,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAPA,C;mFASA,yB;MAAA,oC;MAAA,gC;MAAA,iE;MAAA
,uC;QAKoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UAAsB,IAAI,UAAU,oBAAV,
CAAJ,C;YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAPA,C;;MAaI,OA+3KO,qBAAQ
,CA/3KR,GAAe,IAAf,GAAyB,UAAK,CAAL,C;IACpC,C;;MAMI,OAg4KO,qBAAQ,CAh4KR,GAAe,IAAf,GAA
yB,UAAK,CAAL,C;IACpC,C;;MAMI,OAI4KO,qBAAQ,CAj4KR,GAAe,IAAf,GAAyB,UAAK,CAAL,C;IACpC,
C;;MAMI,OAK4KO,qBAAQ,CAI4KR,GAAe,IAAf,GAAyB,UAAK,CAAL,C;IACpC,C;;MAMI,OAm4KO,qBAA
Q,CAn4KR,GAAe,IAAf,GAAyB,UAAK,CAAL,C;IACpC,C;;MAMI,OAO4KO,qBAAQ,CAp4KR,GAAe,IAAf,GA
AyB,UAAK,CAAL,C;IACpC,C;;MAMI,Oaq4KO,qBAAQ,CAr4KR,GAAe,IAAf,GAAyB,UAAK,CAAL,C;IACp
C,C;;MAMI,OAs4KO,qBAAQ,CAt4KR,GAAe,IAAf,GAAyB,UAAK,CAAL,C;IACpC,C;;MAMI,Oau4KO,qBAA
Q,CAv4KR,GAAe,IAAf,GAAyB,UAAK,CAAL,C;IACpC,C;;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAGB,
cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;;MAMoB,
Q;MAAhB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAA
O,O;;MACrD,OAAO,I;IACX,C;;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,I
AAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;;MAMoB,Q;MAAhB,wBAAgB,SAA
hB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;IA
CX,C;;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,
C;UAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SA
AhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;;MAMoB,Q;MAAhB,
wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACr
D,OAAO,I;IACX,C;;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,
OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;+FAEA,yB;MAAA,oC;MAAA,gC;MAAA,uC;QA
IoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UAAsB,IAAI,UAAU,oBAAV,CAAJ,C;
YAAwB,OAAO,O;;QACrD,OAAO,I;MACX,C;KANA,C;wFAQA,yB;MAAA,8D;MAAA,iD;QAKI,OAaW,SA
S,CAAT,IAAc,SAAS,wBAA3B,GAAAsC,UAAl,KAAJ,CAATc,GAAAsD,aAAa,KAAb,C;MACjE,C;KANA,C;0FAQ
A,yB;MAAA,8D;MAAA,iD;QAKI,OAaW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAAsC,UAAl,KAAJ,CAATc,G
AAAsD,aAAa,KAAb,C;MACjE,C;KANA,C;0FAQA,yB;MAAA,8D;MAAA,iD;QAKI,OAaW,SAAS,CAAT,IAAc,
SAAS,wBAA3B,GAAAsC,UAAl,KAAJ,CAATc,GAAAsD,aAAa,KAAb,C;MACjE,C;KANA,C;0FAQA,yB;MAAA,8
D;MAAA,iD;QAKI,OAaW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAAsC,UAAl,KAAJ,CAATc,GAAAsD,aAAa,K
AAb,C;MACjE,C;KANA,C;0FAQA,yB;MAAA,8D;MAAA,iD;QAKI,OAaW,SAAS,CAAT,IAAc,SAAS,wBAA3
B,GAAAsC,UAAl,KAAJ,CAATc,GAAAsD,aAAa,KAAb,C;MACjE,C;KANA,C;0FAQA,yB;MAAA,8D;MAAA,iD;Q
AKI,OAaW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAAsC,UAAl,KAAJ,CAATc,GAAAsD,aAAa,KAAb,C;MACjE,
C;KANA,C;0FAQA,yB;MAAA,8D;MAAA,iD;QAKI,OAaW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAAsC,UAAl,
KAAJ,CAATc,GAAAsD,aAAa,KAAb,C;MACjE,C;KANA,C;0FAQA,yB;MAAA,8D;MAAA,iD;QAKI,OAaW,SA
AS,CAAT,IAAc,SAAS,wBAA3B,GAAAsC,UAAl,KAAJ,CAATc,GAAAsD,aAAa,KAAb,C;MACjE,C;KANA,C;0FA
QA,yB;MAAA,8D;MAAA,gC;MAAA,iD;QAKI,OAaW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAAsC,UAAl,KA
AJ,CAATc,GAAAsD,uBAAa,KAAb,E;MACjE,C;KANA,C;;MAYI,OAaW,SAAS,CAAT,IAAc,SAAS,wBAA3B,G
AAAsC,UAAl,KAAJ,CAATc,GAAAsD,I;IACjE,C;;MAMI,OAaW,SAAS,CAAT,IAAc,SAAS,0BAA3B,GAAAsC,UA
Al,KAAJ,CAATc,GAAAsD,I;IACjE,C;;MAMI,OAaW,SAAS,CAAT,IAAc,SAAS,0BAA3B,GAAAsC,UAAl,KAAJ,
CAATc,GAAAsD,I;IACjE,C;;MAMI,OAaW,SAAS,CAAT,IAAc,SAAS,0BAA3B,GAAAsC,UAAl,KAAJ,CAATc,G
AAAsD,I;IACjE,C;;MAMI,OAaW,SAAS,CAAT,IAAc,SAAS,0BAA3B,GAAAsC,UAAl,KAAJ,CAATc,GAAAsD,I;IA
CjE,C;;MAMI,OAaW,SAAS,CAAT,IAAc,SAAS,0BAA3B,GAAAsC,UAAl,KAAJ,CAATc,GAAAsD,I;IACjE,C;;MA

/IR,GAAe,IAAf,GAAYB,UAAK,mBAAO,CAAP,IAAL,C;IACpC,C;4FAEA,yB;MAAA,0D;MAAA,+C;MAAA,u
C;QAIkB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,CAAQ,CAAb,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,cAAc,
UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,OAAO,I;MACX,C;KATA,C;4
FAWA,yB;MAAA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,CAAQ,CAAb,W;QA
Ad,OAAc,cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAA
O,O;;QAEEnC,OAAO,I;MACX,C;KATA,C;6FAWA,yB;MAAA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAA
a,SAAR,YAAL,SAAK,CAAQ,CAAb,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,I
AAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,OAAO,I;MACX,C;KATA,C;6FAWA,yB;MAAA,0D;M
AAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,CAAQ,CAAb,W;QAAd,OAAc,cAAd,C;UAA
c,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,OAAO,I;M
ACX,C;KATA,C;6FAWA,yB;MAAA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,C
AAQ,CAAb,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAA
J,C;YAAwB,OAAO,O;;QAEEnC,OAAO,I;MACX,C;KATA,C;6FAWA,yB;MAAA,0D;MAAA,+C;MAAA,uC;QAI
kB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,CAAQ,CAAb,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,cAAc,UAAK
,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,OAAO,I;MACX,C;KATA,C;6FAWA,
yB;MAAA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,CAAQ,CAAb,W;QAAd,OA
Ac,cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;Q
AEnC,OAAO,I;MACX,C;KATA,C;6FAWA,yB;MAAA,0D;MAAA,+C;MAAA,uC;QAIkB,Q;QAAA,OAAa,SAA
R,YAAL,SAAK,CAAQ,CAAb,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,U
AAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,OAAO,I;MACX,C;KATA,C;6FAWA,yB;MAAA,0D;MAAA,+
C;MAAA,oC;MAAA,uC;QAIkB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,CAAQ,CAAb,W;QAAd,OAAc,cAAd,C;
UAAc,uB;UACV,cAAc,UAAK,KAAL,C;UACd,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,OAA
O,I;MACX,C;KATA,C;kFAWA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;MACX,C;
KATA,C;oFAWA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;MACX,C;KATA,C;oFA
WA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;MACX,C;KATA,C;oFAWA,yB;MAA
A,mC;MAAA,gD;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;MACX,C;KATA,C;oFAWA,yB;MAAA,mC;MAAA
,gD;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;MACX,C;KATA,C;oFAWA,yB;MAAA,mC;MAAA,gD;MAAA,4
B;QAQI,OAAO,kBAAO,cAAP,C;MACX,C;KATA,C;oFAWA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QAQI,OA
AO,kBAAO,cAAP,C;MACX,C;KATA,C;oFAWA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QAQI,OAAO,kBAAO,c
AAP,C;MACX,C;KATA,C;oFAWA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;MAC
X,C;KATA,C;;MAkBI,IAmuIO,qBAAQ,CAnuIf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAI,MA
AO,iBAAQ,gBAAR,CAAX,C;IACX,C;;MASI,IA+tIO,qBAAQ,CA3tIf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MA
CV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;IACX,C;;MASI,IA2tIO,qBAAQ,CA3tIf,C;QACI,MAAM,2B
AAuB,iBAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;IACX,C;;MASI,IAutIO,qBAAQ,CAvt
If,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;IACX,C;;MASI
,IAMtIO,qBAAQ,CAntIf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,C
AAX,C;IACX,C;;MASI,IA+sIO,qBAAQ,CA/sIf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAI,MA
AO,iBAAQ,gBAAR,CAAX,C;IACX,C;;MASI,IA2sIO,qBAAQ,CA3sIf,C;QACI,MAAM,2BAAuB,iBAAvB,C;M
ACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;IACX,C;;MASI,IAusIO,qBAAQ,CAvsIf,C;QACI,MAAM,2
BAAuB,iBAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;IACX,C;;MASI,IAmsIO,qBAAQ,C
AnsIf,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,UAAI,MAAO,iBAAQ,gBAAR,CAAX,C;IACX,C;;M
AMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,MAAM,2BAAuB,iBAAvB,C;aACX,C;UAAK,iBAK,
CAAL,C;UAAL,K;gBACQ,MAAM,gCAAYB,kCAAzB,C;;MAHIB,W;IAKJ,C;;MAMiB,IAAN,I;MAAA,QAAM,g
BAAN,C;aACH,C;UAAK,MAAM,2BAAuB,iBAAvB,C;aACX,C;UAAK,iBAK,CAAL,C;UAAL,K;gBACQ,MA
AM,gCAAYB,kCAAzB,C;;MAHIB,W;IAKJ,C;;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,MA
AM,2BAAuB,iBAAvB,C;aACX,C;UAAK,iBAK,CAAL,C;UAAL,K;gBACQ,MAAM,gCAAYB,kCAAzB,C;;MA
HIB,W;IAKJ,C;;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,MAAM,2BAAuB,iBAAvB,C;aACX
,C;UAAK,iBAK,CAAL,C;UAAL,K;gBACQ,MAAM,gCAAYB,kCAAzB,C;;MAHIB,W;IAKJ,C;;MAMiB,IAAN,

B,Q;MAFhB,aAAiB,I;MACjB,YAAY,K;MACZ,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,IAAI,UA
AU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAAW,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CA
AC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;IACX,C;;MAQoB,Q;MAFhB,aAAoB,I;MACpB,YAAY,K;MAC
Z,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YA
AW,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO
,M;IACX,C;;MAQoB,Q;MAFhB,aAAqB,I;MACrB,YAAY,K;MACZ,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAh
B,M;QACI,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAAW,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I
;;;MAGhB,IAAI,CAAC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;IACX,C;;MAQoB,Q;MAFhB,aAAmB,I;M
ACnB,YAAY,K;MACZ,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,IAAI,UAAU,OAAV,CAAJ,C;UA
CI,IAAI,KAAJ,C;YAAW,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAAY,OA
AO,I;MACnB,OAAO,M;IACX,C;;MAQoB,Q;MAFhB,aAAoB,I;MACpB,YAAY,K;MACZ,wBAAgB,SAAhB,gB;
QAAGB,cAAA,SAAhB,M;QACI,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAAW,OAAO,I;UACIB,SA
AS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;IACX,C;;MAQoB,Q;
MAFhB,aAAqB,I;MACrB,YAAY,K;MACZ,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,IAAI,UAAU,
OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAAW,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,
KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;IACX,C;;MAQoB,Q;MAFhB,aAAsB,I;MACtB,YAAY,K;MACZ,w
BAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAAW,
OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;I
ACX,C;;MAQoB,Q;MAFhB,aAAuB,I;MACvB,YAAY,K;MACZ,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M
;QACI,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAAW,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;;M
AGhB,IAAI,CAAC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;IACX,C;iGAEA,yB;MAAA,oC;MAAA,gC;MA
AA,uC;QAMoB,Q;QAFhB,aAAoB,I;QACpB,YAAY,K;QACZ,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SA
AhB,O;UACI,IAAI,UAAU,oBAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,OAAO,I;YACIB,SAAS,O;YACT,QAA
Q,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAAy,OAAO,I;QACnB,OAAO,M;MACX,C;KAFa,C;;MatpFI,IAAI,Eb+qFI
,KAAK,Ca/qFT,CAAJ,C;QACI,cb8qFc,sD;Qa7qFd,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb8qFV,OAAO,oBAAo
B,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CAApB,C;IACX,C;;MajrFI,IAAI,Eb2rFI,KAAK,Ca3rFT,CAAJ,C;
QACI,cb0rFc,sD;QazrFd,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb0rFV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP
,IAAU,EAAC,CAAd,CAApB,C;IACX,C;;Ma7rFI,IAAI,EbusFI,KAAK,CavsFT,CAAJ,C;QACI,cbssFc,sD;QarsFd,
MAAM,gCAAYB,OAAQ,WAAjC,C;;MbssFV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CA
ApB,C;IACX,C;;MazsFI,IAAI,EbmtFI,KAAK,CantFT,CAAJ,C;QACI,cbktFc,sD;QajtFd,MAAM,gCAAYB,OAAQ
,WAAjC,C;;MbktFV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CAApB,C;IACX,C;;MartFI,I
AAI,Eb+tFI,KAAK,Ca/tFT,CAAJ,C;QACI,cb8tFc,sD;Qa7tFd,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb8tFV,OA
AO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CAApB,C;IACX,C;;MajuFI,IAAI,Eb2uFI,KAAK,Ca3uF
T,CAAJ,C;QACI,cb0uFc,sD;QazuFd,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb0uFV,OAAO,sBAAoB,gBAAV,mB
AAO,CAAP,IAAU,EAAC,CAAd,CAApB,C;IACX,C;;Ma7uFI,IAAI,EbusFI,KAAK,CavvFT,CAAJ,C;QACI,cbsvF
c,sD;QarvFd,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbsvFV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EA
Ac,CAAd,CAApB,C;IACX,C;;MazvFI,IAAI,EbmwFI,KAAK,CanwFT,CAAJ,C;QACI,cbkwFc,sD;QajwFd,MAA
M,gCAAYB,OAAQ,WAAjC,C;;MbkwFV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CAApB,
C;IACX,C;;MarwFI,IAAI,Eb+wFI,KAAK,Ca/wFT,CAAJ,C;QACI,cb8wFc,sD;Qa7wFd,MAAM,gCAAYB,OAAQ,
WAAjC,C;;Mb8wFV,OAAO,sBAAoB,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CAApB,C;IACX,C;;MajxFI,I
AAI,Eb2xFI,KAAK,Ca3xFT,CAAJ,C;QACI,cb0xFc,sD;QazxFd,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb0xFV,O
AAO,gBAAgB,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB,C;IACX,C;;Ma7xFI,IAAI,EbuyFI,KAAK,C
avyFT,CAAJ,C;QACI,cbSyFc,sD;QaryFd,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbsyFV,OAAO,kBAAgB,gBAAV
,mBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB,C;IACX,C;;MazyFI,IAAI,EbmzFI,KAAK,CanzFT,CAAJ,C;QACI,c
bkzFc,sD;QajzFd,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbKzFV,OAAO,kBAAgB,gBAAV,mBAAO,CAAP,IAAU
,EAAC,CAAd,CAAhB,C;IACX,C;;MarzFI,IAAI,Eb+zFI,KAAK,Ca/zFT,CAAJ,C;QACI,cb8zFc,sD;Qa7zFd,MAA
M,gCAAYB,OAAQ,WAAjC,C;;Mb8zFV,OAAO,kBAAgB,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB,
C;IACX,C;;Maj0FI,IAAI,Eb20FI,KAAK,Ca30FT,CAAJ,C;QACI,cb00Fc,sD;Qaz0Fd,MAAM,gCAAYB,OAAQ,W

AAjC,C;;Mb00FV,OAAO,kBAAGB,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB,C;IACX,C;;Ma70FI,IAAI,Ebu1FI,KAAK,Cav1FT,CAAJ,C;QACI,cbs1Fc,sD;Qar1Fd,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mbs1FV,OA AO,kBAAGB,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB,C;IACX,C;;Maz1FI,IAAI,Ebm2FI,KAAK,Ca n2FT,CAAJ,C;QACI,cbk2Fc,sD;Qaj2Fd,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mbk2FV,OAAO,kBAAGB,gBAAV, mBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB,C;IACX,C;;Mar2FI,IAAI,Eb+2FI,KAAK,Ca/2FT,CAAJ,C;QACI,cb8 2Fc,sD;Qa72Fd,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb82FV,OAAO,kBAAGB,gBAAV,mBAAO,CAAP,IAAU, EAAC,CAAd,CAAhB,C;IACX,C;;Maj3FI,IAAI,Eb23FI,KAAK,Ca33FT,CAAJ,C;QACI,cb03Fc,sD;Qaz3Fd,MAA M,gCAAYB,OAAQ,WAAjC,C;;Mb03FV,OAAO,kBAAGB,gBAAV,mBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB, C;IACX,C;gGAEA,yB;MAAA,8D;MAAA,4C;MAAA,qD;MAAA,uC;QAMI,iBAAC,wBAAd,WAA+B,CAA/B,U; UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf ,OAAO,W;MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,2C;MAAA,qD;MAAA,uC;QAMI,iBAAC,wBAAd, WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CA AR,IAAL,C;;;QAGf,OAAO,W;MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,4C;MAAA,qD;MAAA,uC;QA MI,iBAAC,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO, gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,W;MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,4C;MAAA ,qD;MAAA,uC;QAMI,iBAAC,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CA AL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,W;MACX,C;KAZA,C;kGAcA,yB;MAAA,8 D;MAAA,4C;MAAA,qD;MAAA,uC;QAMI,iBAAC,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UAA K,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,W;MACX,C;KAZA,C;k GAcA,yB;MAAA,8D;MAAA,4C;MAAA,qD;MAAA,uC;QAMI,iBAAC,wBAAd,WAA+B,CAA/B,U;UACI,IAAI, CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,W; MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,4C;MAAA,qD;MAAA,uC;QAMI,iBAAC,wBAAd,WAA+B,C AA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL, C;;;QAGf,OAAO,W;MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,4C;MAAA,qD;MAAA,uC;QAMI,iBAAC, wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,Q AAQ,CAAR,IAAL,C;;;QAGf,OAAO,W;MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,oC;MAAA,4C;MAA A,qD;MAAA,uC;QAMI,iBAAC,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,EA AV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,W;MACX,C;KAZA,C;wFAcA,yB;MAAA, +D;MAAA,uC;QAQiB,Q;QAFb,eAAe,K;QACf,WAAW,gB;QACX,wBAAa,SAAb,gB;UAAa,WAAA,SAAb,M;U ACI,IAAI,QAAJ,C;YACI,IAAK,WAAI,IAAJ,C;eACJ,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,IAAK,WAAI,IAAJ,C;YACL,WAAW,I;;;QAEEnB,OAAO,I;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAAA,uC;QAQiB,Q;Q AFb,eAAe,K;QACf,WAAW,gB;QACX,wBAAa,SAAb,gB;UAAa,WAAA,SAAb,M;UACI,IAAI,QAAJ,C;YACI,IAAK,WAAI,IAAJ,C;eACJ,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,IAAK,WAAI,IAAJ,C;YACL,WAAW,I;;; QAEEnB,OAAO,I;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAAA,uC;QAQiB,Q;QAFb,eAAe,K;QACf,WAA W,gB;QACX,wBAAa,SAAb,gB;UAAa,WAAA,SAAb,M;UACI,IAAI,QAAJ,C;YACI,IAAK,WAAI,IAAJ,C;eACJ, IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,IAAK,WAAI,IAAJ,C;YACL,WAAW,I;;;QAEEnB,OAAO,I;MACX,C; KA hBA,C;0FAkBA,yB;MAAA,+D;MAAA,uC;QAQiB,Q;QAFb,eAAe,K;QACf,WAAW,gB;QACX,wBAAa,SAAb,gB;UAAa,WAAA,SAAb,M;UACI,IAAI,QAAJ,C;YACI,IAAK,WAAI,IAAJ,C;eACJ,IAAI,CAAC,UAAU,IAAV, CAAL,C;YACD,IAAK,WAAI,IAAJ,C;YACL,WAAW,I;;;QAEEnB,OAAO,I;MACX,C;KAhBA,C;0FAkBA,yB;MA AA,+D;MAAA,uC;QAQiB,Q;QAFb,eAAe,K;QACf,WAAW,gB;QACX,wBAAa,SAAb,gB;UAAa,WAAA,SAAb, M;UACI,IAAI,QAAJ,C;YACI,IAAK,WAAI,IAAJ,C;eACJ,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,IAAK,W AAI,IAAJ,C;YACL,WAAW,I;;;QAEEnB,OAAO,I;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAAA,uC;QAQi B,Q;QAFb,eAAe,K;QACf,WAAW,gB;QACX,wBAAa,SAAb,gB;UAAa,WAAA,SAAb,M;UACI,IAAI,QAAJ,C;Y ACI,IAAK,WAAI,IAAJ,C;eACJ,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,IAAK,WAAI,IAAJ,C;YACL,WAA W,I;;;QAEEnB,OAAO,I;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAAA,uC;QAQiB,Q;QAFb,eAAe,K;QACf, WAAW,gB;QACX,wBAAa,SAAb,gB;UAAa,WAAA,SAAb,M;UACI,IAAI,QAAJ,C;YACI,IAAK,WAAI,IAAJ,C; eACJ,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,IAAK,WAAI,IAAJ,C;YACL,WAAW,I;;;QAEEnB,OAAO,I;MA CX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAAA,uC;QAQiB,Q;QAFb,eAAe,K;QACf,WAAW,gB;QACX,wBAA

a,SAAb,gB;UAAa,WAAA,SAAb,M;UACI,IAAI,QAAJ,C;YACI,IAAK,WAAI,IAAJ,C;eACJ,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,IAAK,WAAI,IAAJ,C;YACL,WAAW,I;;;QAEbB,OAAO,I;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAAA,oC;MAAA,gC;MAAA,uC;QAQiB,Q;QAFb,eAAe,K;QACf,WAAW,gB;QACX,wBAAa,SAAb,gB;UAAa,WAAb,UAAa,SAAb,O;UACI,IAAI,QAAJ,C;YACI,IAAK,WAAI,iBAAJ,C;eACJ,IAAI,CAAC,UAAU,iBAAV,CAAL,C;YACD,IAAK,WAAI,iBAAJ,C;YACL,WAAW,I;;;QAEbB,OAAO,I;MACX,C;KAhBA,C;kFAkBA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAS,gB;QAiaA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAjaU,SAiaN,CAAU,OAAV,CAAJ,C;YAAwB,WAAy,WAAI,OAAJ,C;;QAja1D,OAKaO,W;MAjaX,C;KALA,C;oFAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAS,gB;QAkaA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAlaa,SAkaT,CAAU,OAAV,CAAJ,C;YAAwB,WAAy,WAAI,OAAJ,C;;QAla1D,OAmoO,W;MAlaX,C;KALA,C;oFAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAS,gB;QAmaA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IANac,SAmaV,CAAU,OAAV,CAAJ,C;YAAwB,WAAy,WAAI,OAAJ,C;;QAna1D,OAOaO,W;MAnaX,C;KALA,C;oFAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAS,gB;QAoaA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IApaY,SAoaR,CAAU,OAAV,CAAJ,C;YAAwB,WAAy,WAAI,OAAJ,C;;QApa1D,OaqaO,W;MApaX,C;KALA,C;oFAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAS,gB;QAqaA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAraa,SAqaT,CAAU,OAAV,CAAJ,C;YAAwB,WAAy,WAAI,OAAJ,C;;QARA1D,OAsaO,W;MARA,X,C;KALA,C;oFAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAS,gB;QAsaA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAtac,SAsaV,CAAU,OAAV,CAAJ,C;YAAwB,WAAy,WAAI,OAAJ,C;;QAta1D,OaUaO,W;MAtaX,C;KALA,C;oFAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAS,gB;QAuaA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAvae,SAuaX,CAAU,OAAV,CAAJ,C;YAAwB,WAAy,WAAI,OAAJ,C;;QAvA1D,OAWaO,W;MAVaX,C;KALA,C;oFAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAS,gB;QAWaA,Q;QAAhB,iD;UAAgB,cAAhB,e;UAAsB,IAxagB,SAwaZ,CAAU,OAAV,CAAJ,C;YAAwB,WAAy,WAAI,OAAJ,C;;QAXa1D,OAYaO,W;MAXaX,C;KALA,C;oFAOA,yB;MAAA,+D;MAYaA,oC;MAAA,gC;MAzaA,uC;QAIW,kBAAS,gB;QAYaA,Q;QAAhB,iD;UAAgB,cAAhB,0B;UAAsB,IAzaa,SAyaT,CAAU,oBAAV,CAAJ,C;YAAwB,WAAy,WAAI,oBAAJ,C;;QAZa1D,OA0aO,W;MAzaX,C;KALA,C;gGAOA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAGB,gB;QAw8OV,gB;QADb,YAAy,C;QACZ,iD;UAAa,WAAb,e;UAt3OI,IAIFmC,SAkF/B,EAs3OkB,cAt3OIB,EAs3OkB,sBAAt3OIB,WAs3O2B,IAt3O3B,CAAJ,C;YAA2C,sBAAs3OZ,IAt3OY,C;;QAIF/C,OAoFO,W;MANFX,C;KAPA,C;kGASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAGB,gB;QAY8OV,gB;QADb,YAAy,C;QACZ,iD;UAAa,WAAb,e;UAp3OI,IARfS,C,SAqFIC,EAo3OkB,cAp3OIB,EAo3OkB,sBAp3OIB,WAO3O2B,IAp3O3B,CAAJ,C;YAA2C,sBAo3OZ,IAp3OY,C;;QARF/C,OAuFO,W;MATFX,C;KAPA,C;kGASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAGB,gB;QA08OV,gB;QADb,YAAy,C;QACZ,iD;UAAa,WAAb,e;UA13OI,IAxFuC,SAwFnC,EAK3OkB,cA13OIB,EAK3OkB,sBA13OIB,WAK3O2B,IA13O3B,CAAJ,C;YAA2C,sBAK3OZ,IA13OY,C;;QAXF/C,OA0FO,W;MAZF,X,C;KAPA,C;kGASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAGB,gB;QA28OV,gB;QADb,YAAy,C;QACZ,iD;UAAa,WAAb,e;UAh3OI,IA3FqC,SA2FjC,EAg3OkB,cAh3OIB,EAg3OkB,sBAh3OIB,WAg3O2B,IAh3O3B,CAAJ,C;YAA2C,sBAg3OZ,IAh3OY,C;;QA3F/C,OA6FO,W;MA5FX,C;KAPA,C;kGASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAGB,gB;QA48OV,gB;QADb,YAAy,C;QACZ,iD;UAAa,WAAb,e;UA92OI,IA9FsC,SA8FIC,EA82OkB,cA92OIB,EA82OkB,sBA92OIB,WA82O2B,IA92O3B,CAAJ,C;YAA2C,sBA82OZ,IA92OY,C;;QA9F/C,OA6GO,W;MA/FX,C;KAPA,C;kGASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAGB,gB;QA68OV,gB;QADb,YAAy,C;QACZ,iD;UAAa,WAAb,e;UA52OI,IAjGuC,SAiGnC,EA42OkB,cA52OIB,EA42OkB,sBA52OIB,WA42O2B,IA52O3B,CAAJ,C;YAA2C,sBA42OZ,IA52OY,C;;QAJG/C,OAmGO,W;MAIGX,C;KAPA,C;kGASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAGB,gB;QA88OV,gB;QADb,YAAy,C;QACZ,iD;UAAa,WAAb,e;UA12OI,IApGwC,SAoGpC,EA02OkB,cA12OIB,EA02OkB,sBA12OIB,WAO2O2B,IA12O3B,CAAJ,C;YAA2C,sBA02OZ,IA12OY,C;;QAPG/C,OAsGO,W;MArGX,C;KAPA,C;kGASA,yB;MAAA,+D;MAAA,uC;QAMW,kBAAGB,gB;QA+8OV,gB;QADb,YAAy,C;QACZ,iD;UAAa,WAAb,e;UAX2OI,IAvGyC,SAuGrC,EAw2OkB,cAx2OIB,EAw2OkB,sBAx2OIB,WAw2O2B,IAx2O3B,CAAJ,C;YAA2C,sBAw2OZ,IAx2OY,C;;QAVG/C,OAYGO,W;MAXGX,C;KAPA,C;kGASA,yB;MAAA,+D;MAYGA,gC;MAS2OA,oC;MA/8OA,uC;QAMW,kBAAGB,gB;QAG9OV,gB;QADb,YAAy,C;QACZ,iD;UAAa,WAAb,0B;UAAmB,eAAO,cAAP,EAAO,sBAAP,S;UAAA,cAAGB,iB;UAt2O/B,IA1GsC,SA0GIC,CAAU,OAAV,EAAiB,OAAjB,CAAJ,C;YAA2C,sBAAI,OAAJ,C;;QA1G/C,OA4GO,W;MA3GX,C;KAPA,C;;MAS4OiB,gB;MADb,YAAy,C;MACZ,iD;QAAa,WAAb,e;QAt3OI,IAAI,WAs3OkB,cAt3OIB,EAs3OkB,sBAAt3OIB,WAs3O2B,IAt3O3B,CAAJ,C;UAA2C,sBAAs3OZ,IAt3OY,C;;MAE/C,OAAO,W;IACX,C;;MA63OiB,gB;MADb,YAAy,

C;MACZ,iD;QAAa,WAAb,e;QAp3OI,IAAI,WAo3OkB,cAp3OIB,EAo3OkB,sBAp3OIB,WAo3O2B,IAp3O3B,CA
AJ,C;UAA2C,sBAo3OZ,IAp3OY,C;;MAE/C,OAAO,W;IACX,C;;MA23OiB,gB;MADb,YAAAY,C;MACZ,iD;QAA
a,WAAb,e;QAl3OI,IAAI,Wak3OkB,cAl3OIB,EAk3OkB,sBAI3OIB,Wak3O2B,IAI3O3B,CAAJ,C;UAA2C,sBAk
3OZ,IAI3OY,C;;MAE/C,OAAO,W;IACX,C;;MAy3OiB,gB;MADb,YAAAY,C;MACZ,iD;QAAa,WAAb,e;QAh3OI,
IAAI,WAg3OkB,cAh3OIB,EAg3OkB,sBAh3OIB,WAg3O2B,IAh3O3B,CAAJ,C;UAA2C,sBAg3OZ,IAh3OY,C;;
MAE/C,OAAO,W;IACX,C;;MAu3OiB,gB;MADb,YAAAY,C;MACZ,iD;QAAa,WAAb,e;QA92OI,IAAI,WA82OkB
,cA92OIB,EA82OkB,sBA92OIB,WA82O2B,IA92O3B,CAAJ,C;UAA2C,sBA82OZ,IA92OY,C;;MAE/C,OAAO,W
;IACX,C;;MAq3OiB,gB;MADb,YAAAY,C;MACZ,iD;QAAa,WAAb,e;QA52OI,IAAI,WA42OkB,cA52OIB,EA42O
kB,sBA52OIB,WA42O2B,IA52O3B,CAAJ,C;UAA2C,sBA42OZ,IA52OY,C;;MAE/C,OAAO,W;IACX,C;;MAM3
OiB,gB;MADb,YAAAY,C;MACZ,iD;QAAa,WAAb,e;QA12OI,IAAI,WA02OkB,cA12OIB,EA02OkB,sBA12OIB,W
A02O2B,IA12O3B,CAAJ,C;UAA2C,sBA02OZ,IA12OY,C;;MAE/C,OAAO,W;IACX,C;;MAi3OiB,gB;MADb,YA
AY,C;MACZ,iD;QAAa,WAAb,e;QAx2OI,IAAI,WAw2OkB,cAx2OIB,EAw2OkB,sBAx2OIB,WAw2O2B,IAx2O3
B,CAAJ,C;UAA2C,sBAw2OZ,IAx2OY,C;;MAE/C,OAAO,W;IACX,C;;sGAEA,yB;MAAA,gC;MA52OA,oC;Mat
2OA,oD;QA62OiB,gB;QADb,YAAAY,C;QACZ,iD;UAAa,WAAb,0B;UAAmB,eAAO,cAAP,EAAO,sBAAP,S;UA
AA,cAAgB,iB;Uat2O/B,IAAI,UAAU,OAAV,EAAiB,OAAjB,CAAJ,C;YAA2C,sBAAI,OAAJ,C;;QAE/C,OAAO,
W;MACX,C;KAVA,C;sGAYA,yB;MAAA,+D;MAAA,sC;QAIW,kBAAmB,gB;QAOV,Q;QAAhB,iD;UAAgB,cA
AhB,e;UAAsB,IAAI,YAAJ,C;YAAkB,WAAY,WAAI,OAAJ,C;;QAPpD,OAQO,W;MAPX,C;KALA,C;;MAW0B,
Q;MAAhB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,IAAI,YAAJ,C;UAAkB,WAAY,WAAI,OAA
J,C;;MACpD,OAAO,W;IACX,C;wFAEA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAY,gB;QA8EH,Q;QAAhB,iD;U
AAgB,cAAhB,e;UAAsB,IAAI,CA9ES,SA8ER,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QA9E3
D,OA+EO,W;MA9EX,C;KALA,C;0FAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAY,gB;QA+EH,Q;QAAhB,iD;
UAAgB,cAAhB,e;UAAsB,IAAI,CA/EY,SA+EX,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QA/E
3D,OAGFO,W;MA/EX,C;KALA,C;0FAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAY,gB;QAghFH,Q;QAAhB,iD;
UAAgB,cAAhB,e;UAAsB,IAAI,CAhFa,SAgFZ,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAhF
3D,OAIfo,W;MAhFX,C;KALA,C;0FAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAY,gB;QAIhFH,Q;QAAhB,iD;
UAAgB,cAAhB,e;UAAsB,IAAI,CAjFW,SAiFV,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAJF
3D,OAKFO,W;MAjFX,C;KALA,C;0FAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAY,gB;QAKFH,Q;QAAhB,iD;
UAAgB,cAAhB,e;UAAsB,IAAI,CAIFy,SAkFX,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAI
F3D,OAmFO,W;MAIFX,C;KALA,C;0FAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAY,gB;QAmFH,Q;QAAhB,iD
;UAAgB,cAAhB,e;UAAsB,IAAI,CAAnFa,SAmFZ,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAn
F3D,OAoFO,W;MAoFX,C;KALA,C;0FAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAY,gB;QAoFH,Q;QAAhB,i
D;UAAgB,cAAhB,e;UAAsB,IAAI,CAPfC,SAoFb,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAP
F3D,OaqFO,W;MApFX,C;KALA,C;0FAOA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAY,gB;QAqFH,Q;QAAhB,i
D;UAAgB,cAAhB,e;UAAsB,IAAI,CArFe,SAqFd,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAR
F3D,OAsFO,W;MARFX,C;KALA,C;0FAOA,yB;MAAA,+D;MAsFA,oC;MAAA,gC;MATfA,uC;QAIW,kBAAY,g
B;QAsFH,Q;QAAhB,iD;UAAgB,cAAhB,0B;UAAsB,IAAI,CAtFY,SAsFX,CAAU,oBAAV,CAAL,C;YAAyB,WA
AY,WAAI,oBAAJ,C;;QAtF3D,OAUfo,W;MatFX,C;KALA,C;;MAWI,OAAO,2BAAGB,gBAAhB,C;IACX,C;;M
AMoB,Q;MAAhB,wBAAGB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,IAAI,eAAJ,C;UAAqB,WAAY,WAA
I,OAAJ,C;;MACvD,OAAO,W;IACX,C;;MAMoB,Q;MAAhB,wBAAGB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;Q
AAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;IACX,C;;MA
MoB,Q;MAAhB,wBAAGB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;
UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;IACX,C;;MAMoB,Q;MAAhB,wBAAGB,SAAhB,gB;QAAG
B,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OA
AO,W;IACX,C;;MAMoB,Q;MAAhB,wBAAGB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAA
U,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;IACX,C;;MAMoB,Q;MAAhB,wBAAGB
,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OA
AJ,C;;MAC3D,OAAO,W;IACX,C;;MAMoB,Q;MAAhB,wBAAGB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB
,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;IACX,C;;MAMoB,Q;

X,C;;MASKB,Q;MAHd,WAAmB,wBAAR,OAAQ,EAAwB,EAAxB,C;MACnB,IAAI,SAAQ,CAAZ,C;QAAe,OA
AO,W;MAcTb,WAAW,iBAaKb,IAAlB,C;MACG,yB;MAAd,OAAc,cAAd,C;QAAc,uB;QACV,IAAK,WAAI,UA
Al,KAAJ,CAAJ,C;;MAET,OAAO,I;IACX,C;;MASKB,Q;MAHd,WAAmB,wBAAR,OAAQ,EAAwB,EAAxB,C;M
ACnB,IAAI,SAAQ,CAAZ,C;QAAe,OAAO,W;MAcTb,WAAW,iBAAmB,IAAnB,C;MACG,yB;MAAd,OAAc,cA
Ad,C;QAAc,uB;QACV,IAAK,WAAI,UAAI,KAAJ,CAAJ,C;;MAET,OAAO,I;IACX,C;;MASKB,Q;MAHd,WAAm
B,wBAAR,OAAQ,EAAwB,EAAxB,C;MACnB,IAAI,SAAQ,CAAZ,C;QAAe,OAAO,W;MAcTb,WAAW,iBAAgB
,IAAhB,C;MACG,yB;MAAd,OAAc,cAAd,C;QAAc,uB;QACV,IAAK,WAAI,sBAAl,KAAJ,EAAJ,C;;MAET,OAA
O,I;IACX,C;;MAQwB,UACT,M;MAHX,aAAa,aAAa,SAAb,EAAmB,OAAQ,KAA3B,C;MACb,kBAaKb,C;MAC
E,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QACHB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAAK,WAAL,C;
;MAE5B,OAAO,M;IACX,C;;MAQwB,UACT,M;MAHX,aAAa,cAAU,OAAQ,KAAIB,C;MACb,kBAaKb,C;MAC
E,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QACHB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAAK,WAAL,C;
;MAE5B,OAAO,M;IACX,C;;MAQwB,UACT,M;MAHX,aAAa,eAAW,OAAQ,KAAAnB,C;MACb,kBAaKb,C;MA
CE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QACHB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAAK,WAAL,
C;;MAE5B,OAAO,M;IACX,C;;MAQwB,UACT,M;MAHX,aAAa,eAAS,OAAQ,KAAjB,C;MACb,kBAaKb,C;M
ACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QACHB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAAK,WAAL,
C;;MAE5B,OAAO,M;IACX,C;;MAQwB,UACT,M;MAHX,aAAa,iBAAU,OAAQ,KAAIB,C;MACb,kBAaKb,C;
MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QACHB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAAK,WAAL,
C;;MAE5B,OAAO,M;IACX,C;;MAQwB,UACT,M;MAHX,aAAa,iBAAW,OAAQ,KAAAnB,C;MACb,kBAaKb
,C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QACHB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAAK,
WAAL,C;;MAE5B,OAAO,M;IACX,C;;MAQwB,UACT,M;MAHX,aAAa,iBAAY,OAAQ,KAApB,C;MACb,kBA
AkB,C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QACHB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UAA
K,WAAL,C;;MAE5B,OAAO,M;IACX,C;;MAQwB,UACT,M;MAHX,aAAa,oBAAa,OAAQ,KAArB,C;MACb,kB
AAkB,C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QACHB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,UA
AK,WAAL,C;;MAE5B,OAAO,M;IACX,C;;MAQwB,UACT,M;MAHX,aAAa,iBAAU,OAAQ,KAAIB,C;MACb,k
BAaKb,C;MACE,yB;MAApB,OAAoB,cAApB,C;QAAoB,6B;QACHB,OAAO,oBAAP,EAAO,4BAAP,YAAwB,U
AAK,WAAL,C;;MAE5B,OAAO,M;IACX,C;;MAMI,IAAI,OAAQ,UAAZ,C;QAAuB,OAAO,yBAAY,CAAZ,EAA
e,CAAF,C;MAC9B,OAAO,yBAAY,OAAQ,MAApB,EAA2B,OAAQ,aAAR,GAAuB,CAAvB,IAA3B,C;IACX,C;;
MAMI,IAAI,OAAQ,UAAZ,C;QAAuB,OAAO,cAAU,CAAV,C;MAC9B,OAAO,yBAAY,OAAQ,MAApB,EAA2B
,OAAQ,aAAR,GAAuB,CAAvB,IAA3B,C;IACX,C;;MAMI,IAAI,OAAQ,UAAZ,C;QAAuB,OAAO,eAAW,CAAX,
C;MAC9B,OAAO,yBAAY,OAAQ,MAApB,EAA2B,OAAQ,aAAR,GAAuB,CAAvB,IAA3B,C;IACX,C;;MAMI,I
AAI,OAAQ,UAAZ,C;QAAuB,OAAO,eAAS,CAAT,C;MAC9B,OAAO,yBAAY,OAAQ,MAApB,EAA2B,OAAQ,
aAAR,GAAuB,CAAvB,IAA3B,C;IACX,C;;MAMI,IAAI,OAAQ,UAAZ,C;QAAuB,OAAO,iBAAU,CAAV,C;MA
C9B,OAAO,yBAAY,OAAQ,MAApB,EAA2B,OAAQ,aAAR,GAAuB,CAAvB,IAA3B,C;IACX,C;;MAMI,IAAI,O
AAQ,UAAZ,C;QAAuB,OAAO,iBAAW,CAAX,C;MAC9B,OAAO,yBAAY,OAAQ,MAApB,EAA2B,OAAQ,aAA
R,GAAuB,CAAvB,IAA3B,C;IACX,C;;MAMI,IAAI,OAAQ,UAAZ,C;QAAuB,OAAO,iBAAY,CAAZ,C;MAC9B,
OAAO,yBAAY,OAAQ,MAApB,EAA2B,OAAQ,aAAR,GAAuB,CAAvB,IAA3B,C;IACX,C;;MAMI,IAAI,OAAQ,
UAAZ,C;QAAuB,OAAO,oBAAa,CAAb,C;MAC9B,OAAO,0BAAY,OAAQ,MAApB,EAA2B,OAAQ,aAAR,GAA
uB,CAAvB,IAA3B,C;IACX,C;;MAMI,IAAI,OAAQ,UAAZ,C;QAAuB,OAAO,iBAAU,CAAV,C;MAC9B,OAAO,
0BAAY,OAAQ,MAApB,EAA2B,OAAQ,aAAR,GAAuB,CAAvB,IAA3B,C;IACX,C;;MAgBiB,UACL,M;MaxgIR,
IAAI,EbigII,KAAK,CajgIT,CAAJ,C;QACI,cbggIc,sD;Qa//Hd,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbggIV,IAAI
,MAAK,CAAT,C;QAAy,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAAe,OAAO,iB;MAcTb,IAAI,MAAK,CA
AT,C;QAAy,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAy,C;MACZ,WAAW,iBAAa,CAAb,C;MACX,
wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAI,qDAAW,CAAF,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;
MAET,OAAO,I;IACX,C;;MAgBiB,UACL,M;Ma9hIR,IAAI,EbuhII,KAAK,CavhIT,CAAJ,C;QACI,cbshIc,sD;Qar
hId,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbshIV,IAAI,MAAK,CAAT,C;QAAy,OAAO,W;MACnB,IAAI,KAAK
,gBAAT,C;QAAe,OAAO,mB;MAcTb,IAAI,MAAK,CAAT,C;QAAy,OAAO,OAAO,UAAK,CAAL,CAAP,C;MA
CnB,YAAy,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAA
I,qDAAW,CAAF,C;UACI,K;QACJ,IAAK,WAAL,IAAJ,C;;MAET,OAAO,I;IACX,C;;MAgBiB,UACL,M;MapjIR,I

AAI,Eb6iII,KAAK,Ca7iIT,CAAJ,C;QACI,Cb4iIc,sD;Qa3iId,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb4iIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAe,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAiB,CAAjB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAI,qDAAW,CAAf,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;MAET,OAAO,I;IACX,C;;MAGBiB,UACL,M;Ma1kIR,IAAI,EbmkII,KAAK,CankIT,CAAJ,C;QACI,CbkkIc,sD;QajkId,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbkkIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAe,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAe,CAAf,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAI,qDAAW,CAAf,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;MAET,OAAO,I;IACX,C;;MAGBiB,UACL,M;MahmIR,IAAI,EbylII,KAAK,CazlIT,CAAJ,C;QACI,CbwlIc,sD;QavlId,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbwlIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAe,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAI,qDAAW,CAAf,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;MAET,OAAO,I;IACX,C;;MAGBiB,UACL,M;MatnIR,IAAI,Eb+mII,KAAK,Ca/mIT,CAAJ,C;QACI,Cb8mIc,sD;Qa7mId,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb8mIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAe,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAiB,CAAjB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAI,qDAAW,CAAf,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;MAET,OAAO,I;IACX,C;;MAGBiB,UACL,M;Ma5oIR,IAAI,EbqoII,KAAK,CaroIT,CAAJ,C;QACI,CbooIc,sD;QanoId,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbooIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAe,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAkB,CAAlB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAI,qDAAW,CAAf,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;MAET,OAAO,I;IACX,C;;MAGBiB,UACL,M;Ma7rII,IAAI,Eb2pII,KAAK,Ca3pIT,CAAJ,C;QACI,Cb0pIc,sD;QazpId,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb0pIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAe,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAmB,CAAnB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,IAAI,qDAAW,CAAf,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;MAET,OAAO,I;IACX,C;;MAGBiB,UACL,M;MaxrIR,IAAI,EbirII,KAAK,CajrIT,CAAJ,C;QACI,CbgrIc,sD;Qa/qId,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbgrIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,gBAAT,C;QAe,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,EAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAgB,CAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAb,UAAa,SAAb,O;QACI,IAAI,qDAAW,CAAf,C;UACI,K;QACJ,IAAK,WAAI,iBAAJ,C;;MAET,OAAO,I;IACX,C;;Ma7rII,IAAI,EbusII,KAAK,CavsIT,CAAJ,C;QACI,CbssIc,sD;QarsId,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbssIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,gB;MACX,IAAI,KAAK,IAAT,C;QAe,OAAO,iB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,OAAO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAa,CAAb,C;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,UAAK,KAAL,CAAJ,C;MACT,OAAO,I;IACX,C;;MahtII,IAAI,Eb0tII,KAAK,Ca1tIT,CAAJ,C;QACI,CbytIc,sD;QaxtId,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbytIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,gB;MACX,IAAI,KAAK,IAAT,C;QAe,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,OAAO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAgB,CAAhB,C;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,UAAK,KAAL,CAAJ,C;MACT,OAAO,I;IACX,C;;ManuII,IAAI,Eb6uII,KAAK,Ca7uIT,CAAJ,C;QACI,Cb4uIc,sD;Qa3uId,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb4uIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,gB;MACX,IAAI,KAAK,IAAT,C;QAe,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,OAAO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAiB,CAAjB,C;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,UAAK,KAAL,CAAJ,C;MACT,OAAO,I;IACX,C;;MatvII,IAAI,EbgwII,KAAK,CahwIT,CAAJ,C;QACI,Cb+vIc,sD;Qa9vId,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb+vIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,gB;MACX,IAAI,KAAK,IAAT,C;QAe,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,OAAO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAe,CAAf,C;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WA

AI,UAAK,KAAL,CAAJ,C;MACT,OAAO,I;IACX,C;;MazwII,IAAI,EbmxII,KAAK,CanxIT,CAAJ,C;QACI,cbkxIc ,sD;QajxId,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbkxIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WA AW,gB;MACX,IAAI,KAAK,IAAT,C;QA Ae,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,U AAK,OAAO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAGB,CAAhB,C;MACX,iBAAc,OAAO,CAAP,IAAd,U AA6B,IAA7B,U;QACI,IAAK,WAAI,UAAK,KAAL,CAAJ,C;MACT,OAAO,I;IACX,C;;Ma5xII,IAAI,Eb5yII,KA AK,CatyIT,CAAJ,C;QACI,cbqyIc,sD;QapyId,MAAM,gCAAYB,OAAQ,WAAjC,C;;MbyqIV,IAAI,MAAK,CAAT, C;QAAY,OAAO,W;MACnB,WAAW,gB;MACX,IAAI,KAAK,IAAT,C;QA Ae,OAAO,mB;MACTB,IAAI,MAAK, CAAT,C;QAAY,OAAO,OAAO,UAAK,OAAO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAiB,CAAjB,C;MAC X,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,UAAK,KAAL,CAAJ,C;MACT,OAAO,I;IA CX,C;;Ma/yII,IAAI,EbyzII,KAAK,CazzIT,CAAJ,C;QACI,cbwzIc,sD;QavzId,MAAM,gCAAYB,OAAQ,WAAjC,C ;;MbwwIV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,gB;MACX,IAAI,KAAK,IAAT,C;QA Ae,O AAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,OAAO,CAAP,IAAL,CAAP,C;MACnB, WAAW,iBAAkB,CAAiB,C;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,UAAK,K AAL,CAAJ,C;MACT,OAAO,I;IACX,C;;Mal0II,IAAI,Eb40II,KAAK,Ca50IT,CAAJ,C;QACI,cb20Ic,sD;Qa10Id,M AAM,gCAAYB,OAAQ,WAAjC,C;;Mb20IV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,gB;MAC X,IAAI,KAAK,IAAT,C;QA Ae,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,UAAK,OAAO, CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAmB,CAAnB,C;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B ,U;QACI,IAAK,WAAI,UAAK,KAAL,CAAJ,C;MACT,OAAO,I;IACX,C;;Mar1III,IAAI,Eb+1III,KAAK,Ca/1IT,CA AJ,C;QACI,cb81Ic,sD;Qa71Id,MAAM,gCAAYB,OAAQ,WAAjC,C;;Mb81IV,IAAI,MAAK,CAAT,C;QAAY,OAA O,W;MACnB,WAAW,gB;MACX,IAAI,KAAK,IAAT,C;QA Ae,OAAO,mB;MACTB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,OAAO,CAAP,IAAL,EAAP,C;MACnB,WAAW,iBAAGB,CAAhB,C;MACX,iBAAc,OA AO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,sBAAK,KAAL,EA AJ,C;MACT,OAAO,I;IACX,C;gGAE A,yB;MAAA,8D;MAAA,4C;MAAA,gD;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAA C,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;QAGf,OAAO,iB;MAC X,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,2C;MAAA,gD;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,CAA/B, U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;QA Gf,OAAO,iB;MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,4C;MAAA,gD;MAAA,uC;QAMI,iBAAc,wBAA d,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,C AAR,IAAL,C;;QAGf,OAAO,iB;MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,4C;MAAA,gD;MAAA,uC;Q AMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAA O,gBAAK,QAAQ,CAAR,IAAL,C;;QAGf,OAAO,iB;MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,4C;MA AAA,gD;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV, CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;QAGf,OAAO,iB;MACX,C;KAZA,C;kGAcA,yB;MAAA ,8D;MAAA,4C;MAAA,gD;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,UA AK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;QAGf,OAAO,iB;MACX,C;KAZA,C; kGAcA,yB;MAAA,8D;MAAA,4C;MAAA,gD;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI, CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;QAGf,OAAO,iB; MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,4C;MAAA,gD;MAAA,uC;QAMI,iBAAc,wBAAd,WAA+B,C AA/B,U;UACI,IAAI,CAAC,UAAU,UAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL, C;;QAGf,OAAO,iB;MACX,C;KAZA,C;kGAcA,yB;MAAA,8D;MAAA,oC;MAAA,4C;MAAA,gD;MAAA,uC;Q AMI,iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,EA AV,CAAL,C;YACI,OAA O,gBAAK,QAAQ,CAAR,IAAL,C;;QAGf,OAAO,iB;MACX,C;KAZA,C;wFAcA,yB;MAAA,+D;MAAA,uC;QAO iB,Q;QADb,WAAW,gB;QACX,wBAaA,SAAb,gB;UAAa,WAAA,SAAb,M;UACI,IAAI,CAAC,UAAU,IAAV,CA AL,C;YACI,K;UACJ,IAAK,WAAI,IAAJ,C;;QAET,OAAO,I;MACX,C;KAbA,C;0FAeA,yB;MAAA,+D;MAAA,u C;QAOiB,Q;QADb,WAAW,gB;QACX,wBAaA,SAAb,gB;UAAa,WAAA,SAAb,M;UACI,IAAI,CAAC,UAAU,IA AV,CAAL,C;YACI,K;UACJ,IAAK,WAAI,IAAJ,C;;QAET,OAAO,I;MACX,C;KAbA,C;0FAeA,yB;MAAA,+D;M AAA,uC;QAOiB,Q;QADb,WAAW,gB;QACX,wBAaA,SAAb,gB;UAAa,WAAA,SAAb,M;UACI,IAAI,CAAC,UA AU,IAAV,CAAL,C;YACI,K;UACJ,IAAK,WAAI,IAAJ,C;;QAET,OAAO,I;MACX,C;KAbA,C;0FAeA,yB;MAAA

Z,IAAP,IAAwB,UAAK,CAAL,C;MAC5B,OAAO,M;IACX,C;;MAMI,IAg/CO,qBAAQ,CAh/Cf,C;QAAe,OAAO,S;MACtB,aAAa,cAAU,gBAAV,C;MACb,gBAAgB,0B;MACHb,aAAU,CAAV,OAAa,SAAb,M;QACI,OAAO,YAAY,CAAZ,IAAP,IAAwB,UAAK,CAAL,C;MAC5B,OAAO,M;IACX,C;;MAMI,IA4+CO,qBAAQ,CA5+Cf,C;QA Ae,OAAO,S;MACtB,aAAa,eAAW,gBAAX,C;MACb,gBAAgB,0B;MACHb,aAAU,CAAV,OAAa,SAAb,M;QACI,OAAO,YAAY,CAAZ,IAAP,IAAwB,UAAK,CAAL,C;MAC5B,OAAO,M;IACX,C;;MAMI,IAw+CO,qBAAQ,Cax +Cf,C;QAAe,OAAO,S;MACtB,aAAa,eAAS,gBAAT,C;MACb,gBAAgB,0B;MACHb,aAAU,CAAV,OAAa,SAAb, M;QACI,OAAO,YAAY,CAAZ,IAAP,IAAwB,UAAK,CAAL,C;MAC5B,OAAO,M;IACX,C;;MAMI,IAo+CO,qB AAQ,Cap+Cf,C;QAAe,OAAO,S;MACtB,aAAa,iBAAU,gBAAV,C;MACb,gBAAgB,0B;MACHb,aAAU,CAAV,O AAa,SAAb,M;QACI,OAAO,YAAY,CAAZ,IAAP,IAAwB,UAAK,CAAL,C;MAC5B,OAAO,M;IACX,C;;MAMI,I Ag+CO,qBAAQ,CAh+Cf,C;QAAe,OAAO,S;MACtB,aAAa,iBAAW,gBAAX,C;MACb,gBAAgB,0B;MACHb,aAA U,CAAV,OAAa,SAAb,M;QACI,OAAO,YAAY,CAAZ,IAAP,IAAwB,UAAK,CAAL,C;MAC5B,OAAO,M;IACX, C;;MAMI,IA49CO,qBAAQ,CA59Cf,C;QAe,OAAO,S;MACtB,aAAa,iBAAy,gBAAZ,C;MACb,gBAAgB,0B;M AChB,aAAU,CAAV,OAAa,SAAb,M;QACI,OAAO,YAAY,CAAZ,IAAP,IAAwB,UAAK,CAAL,C;MAC5B,OAA O,M;IACX,C;;MAMI,IAw9CO,qBAAQ,Cax9Cf,C;QAAe,OAAO,S;MACtB,aAAa,oBAAb,gBAAb,C;MACb,gBA AgB,0B;MACHb,aAAU,CAAV,OAAa,SAAb,M;QACI,OAAO,YAAY,CAAZ,IAAP,IAAwB,UAAK,CAAL,C;MA C5B,OAAO,M;IACX,C;;MAMI,IAo9CO,qBAAQ,Cap9Cf,C;QAAe,OAAO,S;MACtB,aAAa,iBAAU,gBAAV,C; MACb,gBAAgB,0B;MACHb,aAAU,CAAV,OAAa,SAAb,M;QACI,OAAO,YAAY,CAAZ,IAAP,IAAwB,UAAK,C AAL,C;MAC5B,OAAO,M;IACX,C;kFAEA,yB;MAAA,oD;Mej4JA,sC;MCjFA,kC;MAAA,oC;;QAAsG,4C;MAAt G,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDiFA,uBAOe,yB;QArEf,8D;;U AqEe,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CAA3B, C;UA4DI,C;QAAA,C;OAAA,C;Mf03Jf,sC;QAMI,IAAI,mBAAO,CAAX,C;UAAC,iDeh4JH,iBfg4JsB,Qeh4JtB,Cf g4JG,E;;MACIB,C;KAPA,C;sGASA,yB;MAAA,oD;Mev3JA,sC;MCpGA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C; MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDoGA,iCAOe,yB;QAxFf,8D;;UAw Fe,uB;YAAU,eAAsB,gB;YAAtB,OA/Ed,cAAc,SA+EgB,CA/EhB,CAAd,EAA2B,SA+EM,CA/EN,CAA3B,C;UA+ EI,C;QAAA,C;OAAA,C;Mfg3Jf,sC;QAMI,IAAI,mBAAO,CAAX,C;UAAC,iDet3JH,2Bfs3JgC,Qet3JhC,Cfs3JG,E; ;MACIB,C;KAPA,C;;MAEi,oBAAS,cAAT,C;IACJ,C;;MAMI,IAAI,mBAAO,CAAX,C;QACI,e;QACA,oB;;IAER, C;;MAMI,IAAI,mBAAO,CAAX,C;QACI,e;QACA,oB;;IAER,C;;MAMI,IAAI,mBAAO,CAAX,C;QACI,e;QACA, oB;;IAER,C;;MAMI,IAAI,mBAAO,CAAX,C;QACI,iB;QACA,oB;;IAER,C;;MAMI,IAAI,mBAAO,CAAX,C;QA CI,e;QACA,oB;;IAER,C;;MAMI,IAAI,mBAAO,CAAX,C;QACI,e;QACA,oB;;IAER,C;;MAMI,IAAI,mBAAO,CA AX,C;QACI,e;QACA,oB;;IAER,C;;MAQI,OAaqB,OAAd,sBAAC,C;IACzB,C;;MAM0B,kBAAf,yB;MAAuB,mB; MAA9B,OAAuC,OiB1/JhC,WjB0/JgC,C;IAC3C,C;;MAM0B,kBAAf,yB;MAAuB,mB;MAA9B,OAAuC,OiBjgKh C,WjBigKgC,C;IAC3C,C;;MAM0B,kBAAf,yB;MAAuB,mB;MAA9B,OAAuC,OiBxgKhC,WjBwgKgC,C;IAC3C, C;;MAM0B,kBAAf,yB;MAAuB,mB;MAA9B,OAAuC,OiB/gKhC,WjB+gKgC,C;IAC3C,C;;MAM0B,kBAAf,yB; MAAuB,mB;MAA9B,OAAuC,OiBthKhC,WjBshKgC,C;IAC3C,C;;MAM0B,kBAAf,yB;MAAuB,mB;MAA9B,O A AuC,OiB7hKhC,WjB6hKgC,C;IAC3C,C;;MAM0B,kBAAf,0B;MAAuB,mB;MAA9B,OAAuC,OiBpiKhC,WjBoi KgC,C;IAC3C,C;;MAQI,IA2uCO,qBAAQ,CA3u Cf,C;QAAe,OAAO,S;MACD,kBAAd,Scr5IiB,Q;Mdq5IK,mB;M AA7B,OiB9iKO,W;IjB+iKX,C;;MAMI,IA2uCO,qBAAQ,CA3u Cf,C;QAAe,OAAO,S;MACD,kBAAd,Scn5IiB,Q; Mdm5IK,iB;MAA7B,OiBtjKO,W;IjBujKX,C;;MAMI,IA2uCO,qBAAQ,CA3u Cf,C;QAAe,OAAO,S;MACD,kBAA d,Scj5IiB,Q;Mdi5IK,iB;MAA7B,OiB9jKO,W;IjB+jKX,C;;MAMI,IA2uCO,qBAAQ,CA3u Cf,C;QAAe,OAAO,S;M ACD,kBAAd,Sc/4IiB,Q;Md+4IK,iB;MAA7B,OiBtkKO,W;IjBukKX,C;;MAMI,IA2uCO,qBAAQ,CA3u Cf,C;QAA e,OAAO,S;MACD,kBAAT,UAAAL,SAAK,C;MAAiB,mB;MAA7B,OiB9kKO,W;IjB+kKX,C;;MAMI,IA2uCO,qB AAQ,CA3u Cf,C;QAAe,OAAO,S;MACD,kBAAd,Sc54IiB,Q;Md44IK,iB;MAA7B,OiBtlKO,W;IjBulKX,C;;MAMI ,IA2uCO,qBAAQ,CA3u Cf,C;QAAe,OAAO,S;MACD,kBAAd,Sc14IiB,Q;Md04IK,iB;MAA7B,OiB9lKO,W;IjB+l KX,C;;MAMI,IAmvCO,qBAAQ,CAnv Cf,C;QAAe,OAAO,S;MACD,kBAAT,UAAAL,SAAK,C;MAAiB,iB;MAA7 B,OiBtmKO,W;IjBumKX,C;;MAQI,IAyqCO,qBAAQ,CAzq Cf,C;QAAe,OAAO,S;MACD,kBAAd,Scv9IiB,Q;Mdu 9IK,sBAAS,cAAT,C;MAA7B,OiBhnKO,W;IjBinKX,C;;MAMI,IAyqCO,qBAAQ,CAzq Cf,C;QAAe,OAAO,S;MA CD,kBAAd,Scr9IiB,Q;Mdq9IK,6B;MAA7B,OiBxnKO,W;IjBynKX,C;;MAMI,IAyqCO,qBAAQ,CAzq Cf,C;QAAe, OAAO,S;MACD,kBAAd,Scn9IiB,Q;Mdm9IK,6B;MAA7B,OiBhoKO,W;IjBioKX,C;;MAMI,IAyqCO,qBAAQ,CA

zqCf,C;QAAe,OAAO,S;MACD,kBAAd,Scj9IiB,Q;Mdi9IK,6B;MAA7B,OiBxoKO,W;IjByoKX,C;;MAMI,IAyqC
O,qBAAQ,CAzqCf,C;QAAe,OAAO,S;MACD,kBAAT,UAAL,SAAK,C;MAAiB,6B;MAA7B,OiBhpKO,W;IjBipK
X,C;;MAMI,IAyqCO,qBAAQ,CAzqCf,C;QAAe,OAAO,S;MACD,kBAAd,Sc98IiB,Q;Md88IK,6B;MAA7B,OiBxp
KO,W;IjBypKX,C;;MAMI,IAyqCO,qBAAQ,CAzqCf,C;QAAe,OAAO,S;MACD,kBAAd,Sc58IiB,Q;Md48IK,6B;
MAA7B,OiBhqKO,W;IjBiqKX,C;;MAMI,IAirCO,qBAAQ,CAjrCf,C;QAAe,OAAO,S;MACD,kBAAT,UAAL,SA
AK,C;MAAiB,6B;MAA7B,OiBxqKO,W;IjByqKX,C;;MAQI,IAumCO,qBAAQ,CAvmCf,C;QAAe,OAAO,S;MAC
D,kBAAd,SczhJiB,Q;MdyhJK,iC;MAA7B,OiBlrKO,W;IjBmrKX,C;sFAEA,yB;MAAA,wD;Me1qKA,sC;MCjFA,k
C;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;
MDiFA,uBAOe,yB;QArEf,8D;;UAqEe,uB;YAAU,eAAsB,gB;YAAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,E
AA2B,SA4DM,CA5DN,CAA3B,C;UA4DI,C;QAAA,C;OAAA,C;MfmqKf,sC;QAMI,OAAO,mDezqKI,iBfyqKiB,
QezqKjB,CfyqKJ,E;MACX,C;KAPA,C;wFASA,yB;MAAA,wD;MenrKA,sC;MCjFA,kC;MAAA,oC;;QAAsG,4C;
MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDiFA,uBAOe,yB;QArEf
,8D;;UAqEe,uB;YAAU,eAAsB,gB;YAAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CA
A3B,C;UA4DI,C;QAAA,C;OAAA,C;Mf4qKf,sC;QAI,OAAO,mDehrKI,iBfgrKiB,QehrKjB,CfgrKJ,E;MACX,C;
KALA,C;wFAOA,yB;MAAA,wD;Me1rKA,sC;MCjFA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C
,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDiFA,uBAOe,yB;QArEf,8D;;UAqEe,uB;YAAU,eAAsB
,gB;YAAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CAA3B,C;UA4DI,C;QAAA,C;OA
AA,C;MfmrKf,sC;QAI,OAAO,mDevrKI,iBfurKiB,QevrKjB,CfurKJ,E;MACX,C;KALA,C;wFAOA,yB;MAAA,w
D;MejsKA,sC;MCjFA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,
C;MAAA,C;MAD5C,kF;MDiFA,uBAOe,yB;QArEf,8D;;UAqEe,uB;YAAU,eAAsB,gB;YAAAtB,OA5Dd,cAAc,SA
4DgB,CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CAA3B,C;UA4DI,C;QAAA,C;OAAA,C;Mf0rKf,sC;QAI,OAA
O,mDe9rKI,iBf8rKiB,Qe9rKjB,Cf8rKJ,E;MACX,C;KALA,C;wFAOA,yB;MAAA,wD;MexsKA,sC;MCjFA,kC;M
AAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDi
FA,uBAOe,yB;QArEf,8D;;UAqEe,uB;YAAU,eAAsB,gB;YAAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2
B,SA4DM,CA5DN,CAA3B,C;UA4DI,C;QAAA,C;OAAA,C;MfisKf,sC;QAI,OAAO,mDersKI,iBfqsKiB,QersKjB
,CfqsKJ,E;MACX,C;KALA,C;wFAOA,yB;MAAA,wD;Me/sKA,sC;MCjFA,kC;MAAA,oC;;QAAsG,4C;MAAtG,
C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDiFA,uBAOe,yB;QArEf,8D;;UA
qEe,uB;YAAU,eAAsB,gB;YAAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CAA3B,C;
UA4DI,C;QAAA,C;OAAA,C;MfwsKf,sC;QAI,OAAO,mDe5sKI,iBf4sKiB,Qe5sKjB,Cf4sKJ,E;MACX,C;KALA,
C;wFAOA,yB;MAAA,wD;MettKA,sC;MCjFA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAA
W,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDiFA,uBAOe,yB;QArEf,8D;;UAqEe,uB;YAAU,eAAsB,gB;Y
AAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CAA3B,C;UA4DI,C;QAAA,C;OAAA,C
;Mf+sKf,sC;QAI,OAAO,mDentKI,iBfmtKiB,QentKjB,CfmtKJ,E;MACX,C;KALA,C;wFAOA,yB;MAAA,wD;Me
7tKA,sC;MCjFA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MA
AA,C;MAD5C,kF;MDiFA,uBAOe,yB;QArEf,8D;;UAqEe,uB;YAAU,eAAsB,gB;YAAAtB,OA5Dd,cAAc,SA4DgB,
CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CAA3B,C;UA4DI,C;QAAA,C;OAAA,C;MfstKf,sC;QAI,OAAO,mDe
1tKI,iBf0tKiB,Qe1tKjB,Cf0tKJ,E;MACX,C;KALA,C;wFAOA,yB;MAAA,wD;MepuKA,sC;MCjFA,kC;MAAA,o
C;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDiFA,uB
AOe,yB;QArEf,8D;;UAqEe,uB;YAAU,eAAsB,gB;YAAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2B,SA4
DM,CA5DN,CAA3B,C;UA4DI,C;QAAA,C;OAAA,C;Mf6tKf,sC;QAI,OAAO,mDejuKI,iBfiuKiB,QejuKjB,CfiuK
J,E;MACX,C;KALA,C;OGAOA,yB;MAAA,wD;MextKA,sC;MCpGA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MA
AA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDoGA,iCAOe,yB;QAxFf,8D;;UAwFe,u
B;YAAU,eAAsB,gB;YAAAtB,OA/Ed,cAAc,SA+EgB,CA/EhB,CAAd,EAA2B,SA+EM,CA/EN,CAA3B,C;UA+EI,
C;QAAA,C;OAAA,C;MfitKf,sC;QAMI,OAAO,mDevtKI,2BfutK2B,QevtK3B,CfutKJ,E;MACX,C;KAPA,C;4GAS
A,yB;MAAA,wD;MejuKA,sC;MCpGA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAA
X,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDoGA,iCAOe,yB;QAxFf,8D;;UAwFe,uB;YAAU,eAAsB,gB;YAAAtB,O
A/Ed,cAAc,SA+EgB,CA/EhB,CAAd,EAA2B,SA+EM,CA/EN,CAA3B,C;UA+EI,C;QAAA,C;OAAA,C;Mf0tKf,s
C;QAI,OAAO,mDe9tKI,2Bf8tK2B,Qe9tK3B,Cf8tKJ,E;MACX,C;KALA,C;4GAOA,yB;MAAA,wD;MexuKA,sC;

OI,OAAO,EAxEA,qBAAQ,CAwER,C;IACX,C;;MAOI,OAAO,EAxEA,qBAAQ,CAwER,C;IACX,C;;MAOI,OAAO,EAxEA,qBAAQ,CAwER,C;IACX,C;;MAOI,OAAO,EAxEA,qBAAQ,CAwER,C;IACX,C;;MAMY,0BAAO,CAAP,I;IAAA,C;;MAMA,0BAAO,CAAP,I;IAAA,C;;MAMA,0BAAO,CAAP,I;IAAA,C;;MAMA,0BAAO,CAAP,I;IAAA,C;;MAMA,0BAAO,CAAP,I;IAAA,C;;MAMA,0BAAO,CAAP,I;IAAA,C;;MAMA,0BAAO,CAAP,I;IAAA,C;;MAMA,0BAAO,CAAP,I;IAAA,C;;MAMA,0BAAO,CAAP,I;IAAA,C;;MAuNkB,wB;QAAW,2BAAK,KAAL,C;MAAX,C;IAAA,C;;MAA1B,OAAO,qBAAa,gBAAb,EAAMb,gCAAnB,C;IACX,C;;MAM2B,wB;QAAW,wBAAK,KAAL,C;MAAX,C;IAAA,C;;MAAvB,OAAO,+BAAU,gBAAV,GAAgB,6BAAhB,C;IACX,C;;MAM2B,wB;QAAW,wBAAK,KAAL,C;MAAX,C;IAAA,C;;MAAvB,OAAO,kBAAU,gBAAV,EAAGb,6BAAhB,C;IACX,C;;MAM6B,wB;QAAW,0BAAK,KAAL,C;MAAX,C;IAAA,C;;MAAzB,OAAO,kCAAY,gBAAZ,GAaKB,+BAAIB,C;IACX,C;;MAM4B,wB;QAAW,yBAAK,KAAL,C;MAAX,C;IAAA,C;;MAAxB,OAAO,kCAAW,gBAAx,GAAiB,8BAAjB,C;IACX,C;;MAM0B,wB;QAAW,uBAAK,KAAL,C;MAAX,C;IAAA,C;;MAAtB,OAAO,gCAAS,gBAAT,GAaE,4BAAf,C;IACX,C;;MAM2B,wB;QAAW,wBAAK,KAAL,C;MAAX,C;IAAA,C;;MAAvB,OAAO,kBAAU,gBAAV,EAAGb,6BAAhB,C;IACX,C;;MAM4B,wB;QAAW,yBAAK,KAAL,C;MAAX,C;IAAA,C;;MAAxB,OAAO,gCAAW,gBAAx,GAAiB,8BAAjB,C;IACX,C;wFA0CA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAaKB,EAAC,EAAD,C;QAC1B,kBAAY,mBAAoB,QAAPB,C;QA+kBH,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAhLB8C,SAGlB/B,CAAU,OAAV,C;UO50NnB,wBAAL,IAAK,MAAT,EAAGb,IAAK,OAARb,C;;QP4vMA,OAklBO,W;MAjlBX,C;KAXA,C;0FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAaKB,EAAC,EAAD,C;QAC1B,kBAAY,mBAAoB,QAAPB,C;QA+kBH,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAhLB8C,SAGlB/B,CAAU,OAAV,C;UOz1NnB,wBAAL,IAAK,MAAT,EAAGb,IAAK,OAARb,C;;QPpywMA,OAklBO,W;MAjlBX,C;KAXA,C;0FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAaKB,EAAC,EAAD,C;QAC1B,kBAAY,mBAAoB,QAAPB,C;QA+kBH,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAhLB8C,SAGlB/B,CAAU,OAAV,C;UO2NnB,wBAAL,IAAK,MAAT,EAAGb,IAAK,OAARb,C;;QPsxMA,OAklBO,W;MAjlBX,C;KAXA,C;0FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAaKB,EAAC,EAAD,C;QAC1B,kBAAY,mBAAoB,QAAPB,C;QA+kBH,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAhLB8C,SAGlB/B,CAAU,OAAV,C;UOn3NnB,wBAAL,IAAK,MAAT,EAAGb,IAAK,OAARb,C;;QPmyMA,OAklBO,W;MAjlBX,C;KAXA,C;0FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAaKB,EAAC,EAAD,C;QAC1B,kBAAY,mBAAoB,QAAPB,C;QA+kBH,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAhLB8C,SAGlB/B,CAAU,OAAV,C;UOh4NnB,wBAAL,IAAK,MAAT,EAAGb,IAAK,OAARb,C;;QPgzMA,OAklBO,W;MAjlBX,C;KAXA,C;0FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAaKB,EAAC,EAAD,C;QAC1B,kBAAY,mBAAoB,QAAPB,C;QA+kBH,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAhLB8C,SAGlB/B,CAAU,OAAV,C;UO74NnB,wBAAL,IAAK,MAAT,EAAGb,IAAK,OAARb,C;;QP6zMA,OAklBO,W;MAjlBX,C;KAXA,C;0FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAaKB,EAAC,EAAD,C;QAC1B,kBAAY,mBAAoB,QAAPB,C;QA+kBH,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAhLB8C,SAGlB/B,CAAU,OAAV,C;UO15NnB,wBAAL,IAAK,MAAT,EAAGb,IAAK,OAARb,C;;QP00MA,OAklBO,W;MAjlBX,C;KAXA,C;0FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAaKB,EAAC,EAAD,C;QAC1B,kBAAY,mBAAoB,QAAPB,C;QA+kBH,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAhLB8C,SAGlB/B,CAAU,oBAAV,C;UOp7NnB,wBAAL,IAAK,MAAT,EAAGb,IAAK,OAARb,C;;QPo2MA,OAklBO,W;MAjlBX,C;KAXA,C;4FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAaKB,EAAC,EAAD,C;QAC1B,kBAAY,mBAAoB,QAAPB,C;QA+NL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAAy,aAhOoC,WAgOhC,CAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;QAOhb,OAkOO,W;MAjOX,C;KAXA,C;8FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAaKB,EAAC,EAAD,C;QAC1B,kBAAY,mBAAoB,QAAPB,C;QA+NL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAAy,aAjOuC,WaiOnC,CAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;QAjOhB,OAmOO,W;MAIOX,C;KAXA,C;8FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAA

kB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAwB,QAAXB,C;QAiOL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAAY, aAlOwC,WAKOpC,CAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;QAIOhB,OAoOO,W;MAnOX,C;KAXA,C;8FAaA,y B;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAAKB,EAAC,EAAD,C;Q AC1B,kBAAC,mBAAsB,QAAtB,C;QAKOL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAAY,aAnOsC,WAmOIC,C AAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;QAnOhB,OAqOO,W;MApOX,C;KAXA,C;8FAaA,yB;MAAA,0D;MAAA ,yD;MAAA,uE;MAAA,yC;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAA uB,QAaVb,C;QAmOL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAAY,aApOuC,WaOnC,CAAY,OAAZ,CAAJ, EAA0B,OAA1B,C;;QApOhB,OAsoo,W;MArOX,C;KAXA,C;8FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MA AAY,C;QASI,eAAiC,cAAIB,YAAY,gBAAZ,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAwB,QAAXB,C;QAo OL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAAY,aArOwC,WaqOpC,CAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;; QArOhB,OAuOO,W;MAtOX,C;KAXA,C;8FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yC;QASI,eAAiC, cAAIB,YAAY,gBAAZ,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAyB,QAazB,C;QAqOL,Q;QAaHb,iD;UA AgB,cAAhB,e;UACI,WAAY,aAtOyC,WAsOrC,CAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;QAIOhB,OAwoO,W; MAvOX,C;KAXA,C;8FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yC;QASI,eAAiC,cAAIB,YAAY,gBA AZ,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAA0B,QAa1B,C;QAsOL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI ,WAAY,aAvOOC,WaUOtC,CAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;QAvOhB,OAyOO,W;MAxOX,C;KAXA,C; 8FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAY,OA,oC;MAAA,gC;MAzOA,yC;QASI,eAAiC,cAAIB,YAAY,g BAAZ,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAuB,QAaVb,C;QAuOL,Q;QAaHb,iD;UAAgB,cAAhB,0B; UACI,WAAY,aAxOuC,WaWOnC,CAAY,oBAAZ,CAAJ,EAA0B,oBAA1B,C;;QAxOhB,OA0OO,W;MAzOX,C;K AXA,C;8FAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAQI,eAAiC,cAAIB,YAAY,gBAAZ,CAAKB,E AAc,EAAD,C;QAC1B,kBAAC,mBAAoB,QAaPb,C;QAYOL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAAY,aA1 OoC,WA0OhC,CAAY,OAAZ,CAAJ,EA1OiD,cA0OvB,CAAE,OAaf,CAA1B,C;;QAIOhB,OA4OO,W;MA3OX,C ;KAVA,C;8FAYA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAQI,eAAiC,cAAIB,YAAY,gBAAZ,CAAK B,EAAC,EAAD,C;QAC1B,kBAAC,mBAAoB,QAaPb,C;QA2OL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WAAY,a A5OoC,WA4OhC,CAAY,OAAZ,CAAJ,EA5OiD,cA4OvB,CAAE,OAaf,CAA1B,C;;QA5OhB,OA8OO,W;MA7O X,C;KAVA,C;+FAYA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAQI,eAAiC,cAAIB,YAAY,gBAAZ,C AAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAoB,QAaPb,C;QA6OL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI,WA AY,aA9OoC,WA8OhC,CAAY,OAAZ,CAAJ,EA9OiD,cA8OvB,CAAE,OAaf,CAA1B,C;;QA9OhB,OA9PO,W;M A/OX,C;KAVA,C;+FAYA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAQI,eAAiC,cAAIB,YAAY,gBAA Z,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAoB,QAaPb,C;QA+OL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI, WAAY,aAhPoC,WAgPhC,CAAY,OAAZ,CAAJ,EAhPiD,cAgPvB,CAAE,OAaf,CAA1B,C;;QAhPhB,OAkPO,W; MAjPX,C;KAVA,C;+FAYA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAQI,eAAiC,cAAIB,YAAY,gBA AZ,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAoB,QAaPb,C;QAiPL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI, WAAY,aAlPoC,WakPhC,CAAY,OAAZ,CAAJ,EAIPiD,cAkPvB,CAAE,OAaf,CAA1B,C;;QAIPhB,OAoPO,W;M AnPX,C;KAVA,C;+FAYA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAQI,eAAiC,cAAIB,YAAY,gBAA Z,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAoB,QAaPb,C;QAmPL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI, WAAY,aApPoC,WaOphC,CAAY,OAAZ,CAAJ,EApPiD,cAoPvB,CAAE,OAaf,CAA1B,C;;QApPhB,OAsoPO,W; MArPX,C;KAVA,C;+FAYA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAQI,eAAiC,cAAIB,YAAY,gBA AZ,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAoB,QAaPb,C;QAqPL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI ,WAAY,aAtPoC,WAsPhC,CAAY,OAAZ,CAAJ,EAtPiD,cAsPvB,CAAE,OAaf,CAA1B,C;;QAtPhB,OAwoPO,W;M AvPX,C;KAVA,C;+FAYA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAQI,eAAiC,cAAIB,YAAY,gBAA Z,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAoB,QAaPb,C;QAuPL,Q;QAaHb,iD;UAAgB,cAAhB,e;UACI, WAAY,aAxPoC,WawPhC,CAAY,OAAZ,CAAJ,EAxPiD,cAwPvB,CAAE,OAaf,CAA1B,C;;QAxPhB,OA0PO,W; MAzPX,C;KAVA,C;+FAYA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MA0PA,oC;MAAA,gC;MA1PA,yD;QAQI,eA AiC,cAAIB,YAAY,gBAAZ,CAAKB,EAAC,EAAD,C;QAC1B,kBAAC,mBAAoB,QAaPb,C;QAYPL,Q;QAaHb,iD; UAAgB,cAAhB,0B;UACI,WAAY,aAlPoC,WA0PhC,CAAY,oBAAZ,CAAJ,EA1PiD,cA0PvB,CAAE,oBAAf,CA A1B,C;;QA1PhB,OA4PO,W;MA3PX,C;KAVA,C;;MAoBoB,Q;MAAhB,wBAAGB,SAAhB,gB;QAAgB,cAAA,SA AhB,M;QACI,WAAY,aAl,YAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;MAEhB,OAao,W;IACX,C;;MAUoB,Q;M

AAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,EAA0B,OAA1 B,C;;MAEhB,OAAO,W;IACX,C;;MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,W AAy,aAAI,YAAy,OAAZ,CAAJ,EAA0B,OAA1B,C;;MAEhB,OAAO,W;IACX,C;;MAUoB,Q;MAAhB,wBAAgB, SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,EAA0B,OAA1B,C;;MAEhB,OA AO,W;IACX,C;;MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAA Y,OAAZ,CAAJ,EAA0B,OAA1B,C;;MAEhB,OAAO,W;IACX,C;;MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAA gB,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,EAA0B,OAA1B,C;;MAEhB,OAAO,W;IACX,C;; MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,E AA0B,OAA1B,C;;MAEhB,OAAO,W;IACX,C;;MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAh B,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,EAA0B,OAA1B,C;;MAEhB,OAAO,W;IACX,C;IGAEA,yB;MAA A,oC;MAAA,gC;MAAA,sD;QAQoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UACI, WAAy,aAAI,YAAy,oBAAZ,CAAJ,EAA0B,oBAA1B,C;;QAEhB,OAAO,W;MACX,C;KAZA,C;;MAsoB,Q;M AAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,EAA0B,eAAe, OAAf,CAA1B,C;;MAEhB,OAAO,W;IACX,C;;MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB ,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,EAA0B,eAAe,OAAf,CAA1B,C;;MAEhB,OAAO,W;IACX,C;;MAU oB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,EAA0 B,eAAe,OAAf,CAA1B,C;;MAEhB,OAAO,W;IACX,C;;MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA ,SAAhB,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,EAA0B,eAAe,OAAf,CAA1B,C;;MAEhB,OAAO,W;IACX, C;;MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAA J,EAA0B,eAAe,OAAf,CAA1B,C;;MAEhB,OAAO,W;IACX,C;;MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAg B,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,EAA0B,eAAe,OAAf,CAA1B,C;;MAEhB,OAAO, W;IACX,C;;MAUoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAAy,O AAZ,CAAJ,EAA0B,eAAe,OAAf,CAA1B,C;;MAEhB,OAAO,W;IACX,C;;MAUoB,Q;MAAhB,wBAAgB,SAAhB, gB;QAAgB,cAAA,SAAhB,M;QACI,WAAy,aAAI,YAAy,OAAZ,CAAJ,EAA0B,eAAe,OAAf,CAA1B,C;;MAEh B,OAAO,W;IACX,C;MGAEA,yB;MAAA,oC;MAAA,gC;MAAA,sE;QAQoB,Q;QAAhB,wBAAgB,SAAhB,gB;U AAagB,cAAhB,UAAgB,SAAhB,O;UACI,WAAy,aAAI,YAAy,oBAAZ,CAAJ,EAA0B,eAAe,oBAAf,CAA1B,C;;Q AEhB,OAAO,W;MACX,C;KAZA,C;;MAqBoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAC I,WAAe,UAAU,OAAV,C;QO50NnB,wBAAI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;MP80NA,OAAO,W;IACX ,C;;MASoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAe,UAAU,OAAV,C;QOz1Nn B,wBAAI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;MP21NA,OAAO,W;IACX,C;;MASoB,Q;MAAhB,wBAAgB,S AAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAe,UAAU,OAAV,C;QOt2NnB,wBAAI,IAAK,MAAT,EAAGB,IA AK,OAArB,C;;MPw2NA,OAAO,W;IACX,C;;MASoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB, M;QACI,WAAe,UAAU,OAAV,C;QOn3NnB,wBAAI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;MPq3NA,OAAO, W;IACX,C;;MASoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAe,UAAU,OAAV,C; QOh4NnB,wBAAI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;MPk4NA,OAAO,W;IACX,C;;MASoB,Q;MAAhB,w BAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAe,UAAU,OAAV,C;QO74NnB,wBAAI,IAAK,MAAT, EAAGB,IAAK,OAArB,C;;MP+4NA,OAAO,W;IACX,C;;MASoB,Q;MAAhB,wBAAGB,SAAhB,gB;QAAgB,cAA A,SAAhB,M;QACI,WAAe,UAAU,OAAV,C;QO15NnB,wBAAI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;MP45N A,OAAO,W;IACX,C;;MASoB,Q;MAAhB,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,WAAe,UAAU, OAAV,C;QOv6NnB,wBAAI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;MPy6NA,OAAO,W;IACX,C;8FAEA,yB;M AAA,oC;MAAA,gC;MAAA,oD;QAQoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;U ACI,WAAe,UAAU,oBAAV,C;UOp7NnB,wBAAI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;QP7NA,OAAO,W;M ACX,C;KAXA,C;;MAiBiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAAy,WAAI,IAAJ,C;; MAEhB,OAAO,W;IACX,C;;MAMiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAAy,WAAI, IAAJ,C;;MAEhB,OAAO,W;IACX,C;;MAMiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAA Y,WAAI,IAAJ,C;;MAEhB,OAAO,W;IACX,C;;MAMiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SA Ab,M;QACI,WAAy,WAAI,IAAJ,C;;MAEhB,OAAO,W;IACX,C;;MAMiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SA Ab,M;QACI,WAAy,WAAI,IAAJ,C;;MAEhB,OAAO,W;IACX,C;;MAMiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,W

AAA,SAAb,M;QACI,WAAY,WAAL,IAAJ,C;;MAEhB,OAAO,W;IACX,C;;MAMiB,Q;MAAb,wBAAa,SAAb,gB;
QAAa,WAAA,SAAb,M;QACI,WAAY,WAAL,IAAJ,C;;MAEhB,OAAO,W;IACX,C;;MAMiB,Q;MAAb,wBAAa,S
AAb,gB;QAAa,WAAA,SAAb,M;QACI,WAAY,WAAL,IAAJ,C;;MAEhB,OAAO,W;IACX,C;;MAMiB,Q;MAAb,
wBAAa,SAAb,gB;QAAa,WAAb,UAAa,SAAb,O;QACI,WAAY,WAAL,iBAAJ,C;;MAEhB,OAAO,W;IACX,C;;M
AMI,OAAO,wBAAa,eAAW,YAAY,gBAAZ,CAAX,CAAb,C;IACX,C;;MAMI,OAAO,0BAAa,eAAc,YAAY,gBA
AZ,CAAd,CAAb,C;IACX,C;;MAMI,OAAO,0BAAa,eAAe,YAAY,gBAAZ,CAAf,CAAb,C;IACX,C;;MAMI,OAA
O,0BAAa,eAAa,YAAY,gBAAZ,CAAb,CAAb,C;IACX,C;;MAMI,OAAO,0BAAa,eAAc,YAAY,gBAAZ,CAAd,C
AAb,C;IACX,C;;MAMI,OAAO,0BAAa,eAAe,YAAY,gBAAZ,CAAf,CAAb,C;IACX,C;;MAMI,OAAO,0BAAa,e
AAgB,YAAY,gBAAZ,CAAhB,CAAb,C;IACX,C;;MAMI,OAAO,0BAAa,eAAiB,YAAY,gBAAZ,CAAjB,CAAb,C
;IACX,C;;MAMI,OAAO,0BAAa,eAAc,YAAY,gBAAZ,CAAd,CAAb,C;IACX,C;;MAMiB,IAAN,I;MAAA,QAA
M,gBAAN,C;aACH,C;UAAK,kB;UAAL,K;aACA,C;UAAK,cAAO,UAAK,CAAL,CAAP,C;UAAL,K;gBACa,qB
AAL,SAAK,C;UAHV,K;;MAAP,W;IAKJ,C;;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,kB;UA
AL,K;aACA,C;UAAK,cAAO,UAAK,CAAL,CAAP,C;UAAL,K;gBACa,uBAAL,SAAK,C;UAHV,K;;MAAP,W;I
AKJ,C;;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,kB;UAAL,K;aACA,C;UAAK,cAAO,UAAK,
CAAL,CAAP,C;UAAL,K;gBACa,uBAAL,SAAK,C;UAHV,K;;MAAP,W;IAKJ,C;;MAMiB,IAAN,I;MAAA,QAA
M,gBAAN,C;aACH,C;UAAK,kB;UAAL,K;aACA,C;UAAK,cAAO,UAAK,CAAL,CAAP,C;UAAL,K;gBACa,uB
AAL,SAAK,C;UAHV,K;;MAAP,W;IAKJ,C;;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,kB;UA
AL,K;aACA,C;UAAK,cAAO,UAAK,CAAL,CAAP,C;UAAL,K;gBACa,uBAAL,SAAK,C;UAHV,K;;MAAP,W;I
AKJ,C;;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,kB;UAAL,K;aACA,C;UAAK,cAAO,UAAK,
CAAL,CAAP,C;UAAL,K;gBACa,uBAAL,SAAK,C;UAHV,K;;MAAP,W;IAKJ,C;;MAMiB,IAAN,I;MAAA,QAA
M,gBAAN,C;aACH,C;UAAK,kB;UAAL,K;aACA,C;UAAK,cAAO,UAAK,CAAL,CAAP,C;UAAL,K;gBACa,uB
AAL,SAAK,C;UAHV,K;;MAAP,W;IAKJ,C;;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,kB;UA
AL,K;aACA,C;UAAK,cAAO,UAAK,CAAL,CAAP,C;UAAL,K;gBACa,uBAAL,SAAK,C;UAHV,K;;MAAP,W;I
AKJ,C;;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,kB;UAAL,K;aACA,C;UAAK,cAAO,sBAA
K,CAAL,EAAP,C;UAAL,K;gBACa,uBAAL,SAAK,C;UAHV,K;;MAAP,W;IAKJ,C;;MAMI,OAAO,iBA Ae,aAAL
,SAAK,CAAF,C;IACX,C;;MAOiB,Q;MADb,WAAW,iBAAGB,gBAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WA
AA,SAAb,M;QAAMB,IAAK,WAAL,IAAJ,C;;MACxB,OAAO,I;IACX,C;;MAOiB,Q;MADb,WAAW,iBAAiB,gB
AAjB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAMB,IAAK,WAAL,IAAJ,C;;MACxB,OAAO,I;IA
CX,C;;MAOiB,Q;MADb,WAAW,iBA Ae,gBAAf,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAMB,I
AAK,WAAL,IAAJ,C;;MACxB,OAAO,I;IACX,C;;MAOiB,Q;MADb,WAAW,iBAAGB,gBAAhB,C;MACX,wBAA
a,SAAb,gB;QAAa,WAAA,SAAb,M;QAAMB,IAAK,WAAL,IAAJ,C;;MACxB,OAAO,I;IACX,C;;MAOiB,Q;MAD
b,WAAW,iBAAiB,gBAAjB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAMB,IAAK,WAAL,IAAJ,C;
;MACxB,OAAO,I;IACX,C;;MAOiB,Q;MADb,WAAW,iBAAkB,gBAAlB,C;MACX,wBAAa,SAAb,gB;QAAa,W
AAA,SAAb,M;QAAMB,IAAK,WAAL,IAAJ,C;;MACxB,OAAO,I;IACX,C;;MAOiB,Q;MADb,WAAW,iBAAMB,g
BAAnB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAMB,IAAK,WAAL,IAAJ,C;;MACxB,OAAO,I;I
ACX,C;;MAOiB,Q;MADb,WAAW,iBAAGB,gBAAhB,C;MACX,wBAAa,SAAb,gB;QAAa,WAAb,UAAa,SAAb,
O;QAAMB,IAAK,WAAL,iBAAJ,C;;MACxB,OAAO,I;IACX,C;;MAQiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH
,C;UAAK,iB;UAAL,K;aACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;gBACQ,+BAAa,qBAAiB,YAA
Y,gBAAZ,CAAjB,CAAb,C;UAHL,K;;MAAP,W;IAKJ,C;;MAQiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;U
AAK,iB;UAAL,K;aACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;gBACQ,iCAAA,qBAAoB,YAAY,gB
AAZ,CAApB,CAAb,C;UAHL,K;;MAAP,W;IAKJ,C;;MAQiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK
,iB;UAAL,K;aACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;gBACQ,iCAAA,qBAAqB,YAAY,gBAAZ
,CAArB,CAAb,C;UAHL,K;;MAAP,W;IAKJ,C;;MAQiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;U
AAL,K;aACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;gBACQ,iCAAA,qBAAmB,YAAY,gBAAZ,CA
AnB,CAAb,C;UAHL,K;;MAAP,W;IAKJ,C;;MAQiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAA
L,K;aACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;gBACQ,iCAAA,qBAAoB,YAAY,gBAAZ,CAApB,
CAAb,C;UAHL,K;;MAAP,W;IAKJ,C;;MAQiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K;a
ACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;gBACQ,iCAAA,qBAAqB,YAAY,gBAAZ,CAArB,CAA

b,C;UAHL,K;;MAAP,W;IAKJ,C;;MAQiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K;aACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;gBACQ,iCAAA,qBAAsB,YAAY,gBAAZ,CAAtB,CAAb,C;UAHL,K;;MAAP,W;IAKJ,C;;MAQiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K;aACA,C;UAAK,aAAM,UAAK,CAAL,CAAN,C;UAAL,K;gBACQ,iCAAA,qBAAuB,YAAY,gBAAZ,CAAvB,CAAb,C;UAHL,K;;MAAP,W;IAKJ,C;;MAQiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,iB;UAAL,K;aACA,C;UAAK,aAAM,sBAAK,CAAL,EAAN,C;UAAL,K;gBACQ,iCAAA,qBAAoB,YAAY,gBAAZ,CAApB,CAAb,C;UAHL,K;;MAAP,W;IAKJ,C;oFAEA,yB;MAAA,+D;MA+DA,gD;MA/DA,uC;QAIW,kBAAU,gB;QA+DD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAhE6B,SAgEIB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAJEhB,OAmEO,W;MAIEX,C;KALA,C;sFAOA,yB;MAAA,+D;MAmEA,gD;MAnEA,uC;QAIW,kBAAU,gB;QAmED,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WApE6B,SAoEIB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QARehB,OAUeO,W;MATEx,C;KALA,C;sFAOA,yB;MAAA,+D;MAuEA,gD;MAvEA,uC;QAIW,kBAAU,gB;QAUeD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAxE6B,SAwEIB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAZehB,OA2EO,W;MAIEX,C;KALA,C;sFAOA,yB;MAAA,+D;MA2EA,gD;MA3EA,uC;QAIW,kBAAU,gB;QA2ED,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA5E6B,SA4EIB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7EhB,OA+EO,W;MA9EX,C;KALA,C;sFAOA,yB;MAAA,+D;MA+EA,gD;MA/EA,uC;QAIW,kBAAU,gB;QA+ED,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAhF6B,SAgFIB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAJFhB,OAmFO,W;MAIFX,C;KALA,C;sFAOA,yB;MAAA,+D;MAmFA,gD;MAnFA,uC;QAIW,kBAAU,gB;QAmFD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WApF6B,SAoFIB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QARFhB,OAUFO,W;MATFX,C;KALA,C;sFAOA,yB;MAAA,+D;MAuFA,gD;MAvFA,uC;QAIW,kBAAU,gB;QAUFD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WAxF6B,SAwFIB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAZFhB,OA2FO,W;MAIFX,C;KALA,C;sFAOA,yB;MAAA,+D;MA2FA,gD;MA3FA,uC;QAIW,kBAAU,gB;QA2FD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,WA5F6B,SA4FIB,CAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QA7FhB,OA+FO,W;MA9FX,C;KALA,C;sFAOA,yB;MAAA,+D;MA+FA,oC;MAAA,gD;MAAA,gC;MA/FA,uC;QAIW,kBAAU,gB;QA+FD,Q;QAAhB,iD;UAAgB,cAAhB,OB;UACI,WAhG6B,SAgGIB,CAAU,oBAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAJGhB,OAmGO,W;MAIGX,C;KALA,C;uFAOA,yB;MAAA,gD;MAAA,oD;QAIoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,WAAW,UAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KATA,C;0FAWA,yB;MAAA,gD;MAAA,oD;QAIoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,WAAW,UAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KATA,C;0FAWA,yB;MAAA,gD;MAAA,oD;QAIoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,WAAW,UAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KATA,C;0FAWA,yB;MAAA,gD;MAAA,oD;QAIoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,WAAW,UAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KATA,C;0FAWA,yB;MAAA,gD;MAAA,oD;QAIoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,WAAW,UAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KATA,C;0FAWA,yB;MAAA,gD;MAAA,oD;QAIoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UACI,WAAW,UAAU,oBAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KATA,C;oFAWA,yB;MAAA,wE;MAiOA,+D;MAjOA,yC;QASW,kBAAU,oB;QAIoD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAIoID,WAKoVc,CAAY,OAAZ,C;UOhkQP,U;UADP,YPkkQe,WolkQH,WpkkQwB,GolkQxB,C;UACL,IAAL,aAAJ,C;YACH,aPgkQuC,gB;YAA5B,WO/jQX,aP+jQgC,GO/jQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UP4jQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAPoT,OAsOO,W;MArOX,C;KAVA,C;sFAYA,yB;MAAA,wE;MAsoA,+D;MAToA,yC;QASW,kBAAU,oB;QAsOD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAvOoD,WAuO1C,CAAY,OAAZ,C;UOjlQP,U;UADP,YPmlQe,WOnlQH,WpmlQwB,G

OnlQxB,C;UACL,IAAI,AAJ,C;YACH,aPilQuC,gB;YAA5B,WOhlQX,aPglQgC,GOhlQhC,EAAS,MAAT,C;YAC
A,e;;;YAEA,c;;UP6kQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAzOT,OA2OO,W;MA1OX,C;KAVA,C;sFAYA,yB;
MAAA,wE;MA2OA,+D;MA3OA,yC;QASW,kBAAU,oB;QA2OD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA5Oq
D,WA4O3C,CAAY,OAAZ,C;UOlMQP,U;UADP,YPomQe,WOpMQH,WPomQwB,GOpMQxB,C;UACL,IAAI,aA
AJ,C;YACH,aPkmQuC,gB;YAA5B,WOjmQX,aPimQgC,GOjmQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UP8IQ
A,iB;UACA,IAAK,WAAI,OAAJ,C;;QA9OT,OAgPO,W;MA/OX,C;KAVA,C;sFAYA,yB;MAAA,wE;MAGPA,+D;
MAhPA,yC;QASW,kBAAU,oB;QAgPD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UajPmD,WaiPzC,CAAY,OAAZ
,C;UOnnQP,U;UADP,YPqnQe,WOrnQH,WPqnQwB,GOrnQxB,C;UACL,IAAI,AAAJ,C;YACH,aPmnQuC,gB;YA
A5B,WOlMQX,aPknQgC,GOlnQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UP+mQA,iB;UACA,IAAK,WAAI,OAA
J,C;;QAnPT,OAqPO,W;MApPX,C;KAVA,C;sFAYA,yB;MAAA,wE;MAqPA,+D;MARPA,yC;QASW,kBAAU,oB;
QAqPD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAtPoD,WAsP1C,CAAY,OAAZ,C;UOpOQP,U;UADP,YPsoQe,
WotoQH,WPsoQwB,GotoQxB,C;UACL,IAAI,AAAJ,C;YACH,aPooQuC,gB;YAA5B,WOnoQX,aPmoQgC,GOno
QhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPgoQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAxPT,OA0PO,W;MAzP
X,C;KAVA,C;sFAYA,yB;MAAA,wE;MA0PA,+D;MA1PA,yC;QASW,kBAAU,oB;QA0PD,Q;QAAhB,iD;UAAg
B,cAAhB,e;UACI,UA3PqD,WA2P3C,CAAY,OAAZ,C;UOrpQP,U;UADP,YPupQe,WovpQH,WPupQwB,GOvpQ
xB,C;UACL,IAAI,AAAJ,C;YACH,aPqpQuC,gB;YAA5B,WOppQX,aPopQgC,GOppQhC,EAAS,MAAT,C;YACA,
e;;;YAEA,c;;UPipQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QA7PT,OA+PO,W;MA9PX,C;KAVA,C;sFAYA,yB;MA
AA,wE;MA+PA,+D;MA/PA,yC;QASW,kBAAU,oB;QA+PD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UAhQsD,W
AgQ5C,CAAY,OAAZ,C;UOtqQP,U;UADP,YPwqQe,WoxqQH,WPwqQwB,GOxqQxB,C;UACL,IAAI,AAAJ,C;Y
ACH,aPsqQuC,gB;YAA5B,WOrqQX,aPqqQgC,GOrrQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPkqQA,iB;UA
CA,IAAK,WAAI,OAAJ,C;;QAIQT,OAoQO,W;MAnQX,C;KAVA,C;sFAYA,yB;MAAA,wE;MAoQA,+D;MApQ
A,yC;QASW,kBAAU,oB;QAoQD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UArQuD,WaqQ7C,CAAY,OAAZ,C;U
OvrQP,U;UADP,YPyrQe,WozrQH,WPyrQwB,GOzrQxB,C;UACL,IAAI,AAAJ,C;YACH,aPurQuC,gB;YAA5B,W
OtrQX,aPsrQgC,GOtrQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPmrQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAv
QT,OAyQO,W;MAxQX,C;KAVA,C;sFAYA,yB;MAAA,wE;MAyQA,oC;MAAA,+D;MAAA,gC;MAzQA,yC;QA
SW,kBAAU,oB;QAYQD,Q;QAAhB,iD;UAAgB,cAAhB,0B;UACI,UA1QoD,WA0Q1C,CAAY,oBAAZ,C;UOxsQ
P,U;UADP,YP0sQe,W01sQH,WP0sQwB,GO1sQxB,C;UACL,IAAI,AAAJ,C;YACH,aPwsQuC,gB;YAA5B,W0vs
QX,aPusQgC,GOvsQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPosQA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QA5Q
T,OA8QO,W;MA7QX,C;KAVA,C;sFAYA,yB;MAAA,wE;MA8QA,+D;MA9QA,yD;QAUW,kBAAU,oB;QA8QD
,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,UA/QiD,WA+QvC,CAAY,OAAZ,C;UO1tQP,U;UADP,YP4tQe,W05tQH
,WP4tQwB,GO5tQxB,C;UACL,IAAI,AAAJ,C;YACH,aP0tQuC,gB;YAA5B,W0ztQX,aPytQgC,GOztQhC,EAAS,
MAAT,C;YACA,e;;;YAEA,c;;UPstQA,iB;UACA,IAAK,WajRyD,cAiRrD,CAAe,OAAf,CAAJ,C;;QajRT,OAmR
O,W;MAiRX,C;KAXA,C;sFAaA,yB;MAAA,wE;MAmRA,+D;MAnRA,yD;QAUW,kBAAU,oB;QAmRD,Q;QAA
hB,iD;UAAgB,cAAhB,e;UACI,UApRiD,WaorVc,CAAY,OAAZ,C;UO5uQP,U;UADP,YP8uQe,W09uQH,WP8u
QwB,GO9uQxB,C;UACL,IAAI,AAAJ,C;YACH,aP4uQuC,gB;YAA5B,W03uQX,aP2uQgC,GO3uQhC,EAAS,MA
AT,C;YACA,e;;;YAEA,c;;UPwuQA,iB;UACA,IAAK,WatRyD,cAsRrD,CAAe,OAAf,CAAJ,C;;QatRT,OAwRO,
W;MAvRX,C;KAXA,C;uFAaA,yB;MAAA,wE;MAwRA,+D;MAxRA,yD;QAUW,kBAAU,oB;QAwRD,Q;QAAh
B,iD;UAAgB,cAAhB,e;UACI,UazRiD,WaYRvC,CAAY,OAAZ,C;UO9vQP,U;UADP,YPgwQe,WOhwQH,WPgw
QwB,GOhwQxB,C;UACL,IAAI,AAAJ,C;YACH,aP8vQuC,gB;YAA5B,W07vQX,aP6vQgC,GO7vQhC,EAAS,MA
AT,C;YACA,e;;;YAEA,c;;UP0vQA,iB;UACA,IAAK,WA3RyD,cA2RrD,CAAe,OAAf,CAAJ,C;;QA3RT,OA6RO,
W;MA5RX,C;KAXA,C;uFAaA,yB;MAAA,wE;MA6RA,+D;MA7RA,yD;QAUW,kBAAU,oB;QA6RD,Q;QAAhB,
iD;UAAgB,cAAhB,e;UACI,UA9RiD,WA8RvC,CAAY,OAAZ,C;UOhxQP,U;UADP,YPkxQe,W0lxQH,WPkxQw
B,GOlxQxB,C;UACL,IAAI,AAAJ,C;YACH,aPgxQuC,gB;YAA5B,W0/wQX,aP+wQgC,GO/wQhC,EAAS,MAAT,
C;YACA,e;;;YAEA,c;;UP4wQA,iB;UACA,IAAK,WahSyD,cAgSrD,CAAe,OAAf,CAAJ,C;;QAhST,OakSO,W;M
AjSX,C;KAXA,C;uFAaA,yB;MAAA,wE;MAkSA,+D;MAISA,yD;QAUW,kBAAU,oB;QakSD,Q;QAAhB,iD;UA
AgB,cAAhB,e;UACI,UAnSiD,WAmSvC,CAAY,OAAZ,C;UOlyQP,U;UADP,YPoyQe,WopyQH,WPoyQwB,GOpy
QxB,C;UACL,IAAI,AAAJ,C;YACH,aPkyQuC,gB;YAA5B,W0jyQX,aPiyQgC,GOjyQhC,EAAS,MAAT,C;YAC
A,e;;;YAEA,c;;UP8xQA,iB;UACA,IAAK,WArSyD,cAqSrD,CAAe,OAAf,CAAJ,C;;QArST,OAuSO,W;MAtSX,C;

KAXA,C;uFAaA,yB;MAAA,wE;MAuSA,+D;MAvSA,yD;QAUW,kBAAU,oB;QAUdSD,Q;QAAhB,iD;UAAgB,cA
AhB,e;UACI,UAXSiD,WAWSvC,CAAY,OAAZ,C;UOpzQP,U;UADP,YPszQe,WotzQH,WPszQwB,GotzQxB,C;
UACL,IAAI,aAAJ,C;YACH,aPozQuC,gB;YAA5B,WOnzQX,aPmzQgC,GOnzQhC,EAAS,MAAT,C;YACA,e;;;Y
AEA,c;;UPgzQA,iB;UACA,IAAK,WA1SyD,cA0SrD,CAAE,OAAf,CAAJ,C;;QA1ST,OA4SO,W;MA3SX,C;KAX
A,C;uFAaA,yB;MAAA,wE;MA4SA,+D;MA5SA,yD;QAUW,kBAAU,oB;QA4SD,Q;QAAhB,iD;UAAgB,cAAhB,
e;UACI,UA7SiD,WA6SvC,CAAY,OAAZ,C;UO0tQP,U;UADP,YPw0Qe,W0x0QH,WPw0QwB,G0x0QxB,C;UA
CL,IAAI,aAAJ,C;YACH,aPs0QuC,gB;YAA5B,WOr0QX,aPq0QgC,GOr0QhC,EAAS,MAAT,C;YACA,e;;;YAEA,
c;;UPk0QA,iB;UACA,IAAK,WA/SyD,cA+SrD,CAAE,OAAf,CAAJ,C;;QA/ST,OAIto,W;MAhTX,C;KAXA,C;uF
AaA,yB;MAAA,wE;MAiTA,+D;MAjTA,yD;QAUW,kBAAU,oB;QAITD,Q;QAAhB,iD;UAAgB,cAAhB,e;UACI,
UAITiD,WAKTvc,CAAY,OAAZ,C;UOx1QP,U;UADP,YP01Qe,W011QH,WP01QwB,G011QxB,C;UACL,IAAI,
aAAJ,C;YACH,aPw1QuC,gB;YAA5B,W0v1QX,aPu1QgC,G0v1QhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPo1
QA,iB;UACA,IAAK,WApTyD,cAoTrD,CAAE,OAAf,CAAJ,C;;QApTT,OAsTO,W;MArTX,C;KAXA,C;uFAaA,y
B;MAAA,wE;MAStA,oC;MAAA,+D;MAAA,gC;MAtTA,yD;QAUW,kBAAU,oB;QAsTD,Q;QAAhB,iD;UAAgB,
cAAhB,0B;UACI,UAvTiD,WAuTvC,CAAY,oBAAZ,C;UO12QP,U;UADP,YP42Qe,W052QH,WP42QwB,G052
QxB,C;UACL,IAAI,aAAJ,C;YACH,aP02QuC,gB;YAA5B,W0z2QX,aPy2QgC,G0z2QhC,EAAS,MAAT,C;YAC
A,e;;;YAEA,c;;UPS2QA,iB;UACA,IAAK,WAZTyD,cAyTrD,CAAE,oBAAf,CAAJ,C;;QAZTT,OA2TO,W;MA1TX,
C;KAXA,C;wFAaA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAA,SAAhB,
M;UACI,UAAU,YAAY,OAAZ,C;UOhkQP,U;UADP,YPkkQe,W0lkQH,WPkkQwB,G0lkQxB,C;UACL,IAAI,aA
AJ,C;YACH,aPgkQuC,gB;YAA5B,W0jQX,aP+jQgC,G0jQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UP4jQA,iB
;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;MACX,C;KafA,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,
Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOjlQP,U;UADP,YP
mlQe,W0nlQH,WPmlQwB,G0nlQxB,C;UACL,IAAI,aAAJ,C;YACH,aPilQuC,gB;YAA5B,W0hlQX,aPglQgC,G
0hlQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UP6kQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;MAC
X,C;KafA,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAA,SAAh
B,M;UACI,UAAU,YAAY,OAAZ,C;UOlmpQ,U;UADP,YPomQe,W0pmQH,WPomQwB,G0pmQxB,C;UACL,IA
AI,aAAJ,C;YACH,aPkmQuC,gB;YAA5B,W0jmQX,aPimQgC,G0jmQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U
P8lQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;MACX,C;KafA,C;0FAiBA,yB;MAAA,+D;MAAA,sD
;QASoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOnnQP,U;
UADP,YPqnQe,W0rnQH,WPqnQwB,G0rnQxB,C;UACL,IAAI,aAAJ,C;YACH,aPmnQuC,gB;YAA5B,W0lnQX,
aPknQgC,G0lnQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UP+mQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OA
AO,W;MACX,C;KafA,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,c
AAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOp0QP,U;UADP,YPsoQe,W0toQH,WPsoQwB,G0toQxB,C;U
ACL,IAAI,aAAJ,C;YACH,aPooQuC,gB;YAA5B,W0noQX,aPmoQgC,G0noQhC,EAAS,MAAT,C;YACA,e;;;YA
EA,c;;UPgoQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;MACX,C;KafA,C;0FAiBA,yB;MAAA,+D;M
AAA,sD;QASoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOr
pQP,U;UADP,YPupQe,W0vpQH,WPupQwB,G0vpQxB,C;UACL,IAAI,aAAJ,C;YACH,aPppQuC,gB;YAA5B,W
OppQX,aPopQgC,GOppQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPipQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QA
ET,OAAO,W;MACX,C;KafA,C;0FAiBA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAGB,SAAhB,gB;U
AAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAAZ,C;UOtqQP,U;UADP,YPwqQe,W0xqQH,WPwqQwB,G0xq
QxB,C;UACL,IAAI,aAAJ,C;YACH,aPsqQuC,gB;YAA5B,W0rqQX,aPqqQgC,G0rqQhC,EAAS,MAAT,C;YACA
e;;;YAEA,c;;UPkqQA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAET,OAAO,W;MACX,C;KafA,C;0FAiBA,yB;MAA
A,+D;MAAA,sD;QASoB,Q;QAAhB,wBAAGB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAY,OAA
Z,C;UOvrQP,U;UADP,YPyrQe,W0zrQH,WPyrQwB,G0zrQxB,C;UACL,IAAI,aAAJ,C;YACH,aPurQuC,gB;YAA
5B,W0trQX,aPsrQgC,G0trQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPmrQA,iB;UACA,IAAK,WAAI,OAAJ,C;
;QAET,OAAO,W;MACX,C;KafA,C;0FAiBA,yB;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,sD;QASoB,Q;QAAh
B,wBAAGB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UACI,UAAU,YAAY,oBAAZ,C;UOxsQP,U;UADP,
YP0sQe,W01sQH,WP0sQwB,G01sQxB,C;UACL,IAAI,aAAJ,C;YACH,aPwsQuC,gB;YAA5B,W0vsQX,aPusQg
C,G0vsQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPosQA,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;

MACX,C;KAfA,C;0FAiBA,yB;MAAA,+D;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAAY,OAAZ,C;UO1tQP,U;UADP,YP4tQe,WO5tQH,WP4tQwB,GO5tQxB,C;UACL,IAAI,aAAJ,C;YACH,aP0tQuC,gB;YAA5B,WOztQX,aPytQgC,GOztQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPstQA,iB;UACA,IAAK,WAAI,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAAY,OAAZ,C;UO5uQP,U;UADP,YP8uQe,WO9uQH,WP8uQwB,GO9uQxB,C;UACL,IAAI,aAAJ,C;YACH,aP4uQuC,gB;YAA5B,WO3uQX,aP2uQgC,GO3uQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPwuQA,iB;UACA,IAAK,WAAI,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAhBA,C;2FAkBA,yB;MAAA,+D;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAAY,OAAZ,C;UO9vQP,U;UADP,YPgwQe,WOhwQH,WPgwQwB,GOhwQxB,C;UACL,IAAI,aAAJ,C;YACH,aP8vQuC,gB;YAA5B,WO7vQX,aP6vQgC,GO7vQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UP0vQA,iB;UACA,IAAK,WAAI,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAhBA,C;2FAkBA,yB;MAAA,+D;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAAY,OAAZ,C;UOhxQP,U;UADP,YPkxQe,WOlxQH,WPkxQwB,GOlxQxB,C;UACL,IAAI,aAAJ,C;YACH,aPgxQuC,gB;YAA5B,WO/wQX,aP+wQgC,GO/wQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UP4wQA,iB;UACA,IAAK,WAAI,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAhBA,C;2FAkBA,yB;MAAA,+D;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAAY,OAAZ,C;UOlYQP,U;UADP,YPoyQe,WOPYQH,WPoyQwB,GOPYQxB,C;UACL,IAAI,aAAJ,C;YACH,aPkyQuC,gB;YAA5B,WOjyQX,aPiyQgC,GOjyQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UP8xQA,iB;UACA,IAAK,WAAI,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAhBA,C;2FAkBA,yB;MAAA,+D;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAAY,OAAZ,C;UOpzQP,U;UADP,YPszQe,WOTzQH,WPszQwB,GOtzQxB,C;UACL,IAAI,aAAJ,C;YACH,aPozQuC,gB;YAA5B,WOnzQX,aPmzQgC,GOnzQhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPgzQA,iB;UACA,IAAK,WAAI,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAhBA,C;2FAkBA,yB;MAAA,+D;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAAY,OAAZ,C;UOt0QP,U;UADP,YPw0Qe,WOx0QH,WPw0QwB,GOx0QxB,C;UACL,IAAI,aAAJ,C;YACH,aPs0QuC,gB;YAA5B,WOr0QX,aPq0QgC,GOr0QhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPk0QA,iB;UACA,IAAK,WAAI,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAhBA,C;2FAkBA,yB;MAAA,+D;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,UAAU,YAAAY,OAAZ,C;UOx1QP,U;UADP,YP01Qe,WO11QH,WP01QwB,GO11QxB,C;UACL,IAAI,aAAJ,C;YACH,aPw1QuC,gB;YAA5B,WOv1QX,aPu1QgC,GOv1QhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPo1QA,iB;UACA,IAAK,WAAI,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAhBA,C;2FAkBA,yB;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,sE;QAUoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UACI,UAAU,YAAAY,oBAAZ,C;UO12QP,U;UADP,YP42Qe,WO52QH,WP42QwB,GO52QxB,C;UACL,IAAI,aAAJ,C;YACH,aP02QuC,gB;YAA5B,WOz2QX,aPy2QgC,GOz2QhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UPs2QA,iB;UACA,IAAK,WAAI,eAAe,oBAaf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,kC;MAAA,4C;;QAQW,sC;QAAA,8C;MARX,C;MAAA,gE;QASqD,OAAgB,qBAAhB,oBAAgB,C;MAAhB,C;MATrD,oE;QAU4C,gCAAY,OAAZ,C;MAAA,C;MAV5C,gF;MAAA,yC;QAQI,2D;MAIJ,C;KAZA,C;4EAcA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAM,eAAa,gBAAb,C;QA6UA,Q;QAAb,iD;UAAa,WAAb,e;UACI,WAAY,WA9UiB,SA8Ub,CAAU,IAAV,CAAJ,C;;QA9UhB,OA+UO,W;MA9UX,C;KARA,C;8EAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAM,eAAa,gBAAb,C;QA6UA,Q;QAAb,iD;UAAa,WAAb,e;UACI,WAAY,WA9UiB,SA8Ub,CAAU,IAAV,CAAJ,C;;QA9UhB,OA+UO,W;MA9UX,C;KARA,C;8EAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAM,eAAa,gBAAb,C;QA6UA,Q;QAAb,iD;UAAa,WAAb,e;UACI,WAAY,WA9UiB,SA8Ub,CAAU,IAAV,CAAJ,C;;QA9UhB,OA+UO,W;MA9UX,C;KARA,C;8EAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAM,eAAa,gBAAb,C;QA6UA,Q;QAAb,iD;UAAa,WAAb,e;UACI,WAAY,WA9UiB,SA8Ub,CAAU,IAAV,CAAJ,C;;QA9UhB,OA+UO,W;MA9UX,C;KARA,C;8EAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAM,eAAa,gBAAb,C;QA6UA,Q;QAAb,iD;UAAa,WAAb,e;UACI,WAAY,WA9UiB,SA8Ub,CAAU,IAAV,CAAJ,C;;QA9UhB,OA+UO,W;MA9UX,C;KARA,C;8EAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAM,eAAa,gBAAb,C;QA6UA,Q;QAAb,iD;UAAa,WAAb,e;UACI,WAAY,WA9UiB,SA8Ub,CAAU,IAAV,CAAJ,C;

,C;;QA9UhB,OA+UO,W;MA9UX,C;KARA,C;8EAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAM,eAAa,gBAAb,C;QA6UA,Q;QAAb,iD;UAAa,WAAb,e;UACI,WAA,Y,WA9UiB,SA8Ub,CAAU,IAAV,CAAJ,C;;QA9UhB,OA+UO,W;MA9UX,C;KARA,C;8EAUA,yB;MAAA,gE;MA+UA,oC;MAAA,gC;MA/UA,uC;QAOW,kBAAM,eAAa,gBAAb,C;QA6UA,Q;QAAb,iD;UAAa,WAAb,0B;UACI,WAA,Y,WA9UiB,SA8Ub,CAAU,iBAAV,CAAJ,C;;QA9UhB,OA+UO,W;MA9UX,C;KARA,C;0FAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAa,eAAa,gBAAb,C;QA9HP,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,e;UACI,WAA,Y,WAjHwB,SaiHpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAjHhB,OakHO,W;MAjHX,C;KARA,C;4FAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAa,eAAa,gBAAb,C;QAmHP,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,e;UACI,WAA,Y,WApHwB,SAoHpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QApHhB,OAqHO,W;MApHX,C;KARA,C;4FAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAa,eAAa,gBAAb,C;QAsHP,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,e;UACI,WAA,Y,WAvHwB,SAuHpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAvHhB,OAwhO,W;MAvHX,C;KARA,C;4FAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAa,eAAa,gBAAb,C;QAYHP,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,e;UACI,WAA,Y,WA1HwB,SA0HpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA1HhB,OA2HO,W;MA1HX,C;KARA,C;4FAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAa,eAAa,gBAAb,C;QA4HP,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,e;UACI,WAA,Y,WA7HwB,SA6HpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA7HhB,OA8HO,W;MA7HX,C;KARA,C;2FAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAa,eAAa,gBAAb,C;QA+HP,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,e;UACI,WAA,Y,WAhIwB,SAGIpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAhIhB,OAiIO,W;MAhIX,C;KARA,C;4FAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAa,eAAa,gBAAb,C;QAKIP,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,e;UACI,WAA,Y,WAnIwB,SAmIpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAnIhB,OAoIO,W;MAnIX,C;KARA,C;4FAUA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAa,eAAa,gBAAb,C;QAqIP,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,e;UACI,WAA,Y,WAtIwB,SAsIpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAtIhB,OAuIO,W;MAtIX,C;KARA,C;4FAUA,yB;MAAA,gE;MAuIA,oC;MAAA,gC;MAvIA,uC;QAOW,kBAAa,eAAa,gBAAb,C;QAwIP,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,0B;UACI,WAA,Y,WazIwB,SAyIpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,iBAAnB,CAAJ,C;;QAzIhB,OA0IO,W;MAzIX,C;KARA,C;wGAUA,yB;MAAA,+D;MAAA,uC;QAOW,kBAAoB,gB;Qao2Dd,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,e;UA11DmC,U;UAAA,cAVQ,SAUR,EA01DT,cA11DS,EA01DT,sBA11DS,WA01DA,IA11DA,W;YAA6C,6B;;;QAVhF,OAwo,W;MAVX,C;KARA,C;4GAUA,yB;MAAA,oD;Qai2DiB,gB;QADb,YAAY,C;QACZ,iD;UAAa,WAAb,e;UA11DmC,U;UAAA,yBA01DT,cA11DS,EA01DT,sBA11DS,WA01DA,IA11DA,W;YAA6C,6B;;;QACHF,OAo,W;MACX,C;KATA,C;;MAMBiB,UACiB,M;MAF9B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAA,Y,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAUiB,UACiB,M;MAF9B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAA,Y,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAUiB,UACiB,M;MAF9B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAA,Y,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAUiB,UACiB,M;MAF9B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAA,Y,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAUiB,UACiB,M;MAF9B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAA,Y,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;MACHB,OAAO,W;IACX,C;gGAEA,yB;MAAA,oC;MAAA,gC;MAAA,oD;QAQiB,UACiB,M;QAF9B,YAAY,C;QACZ,wBAAa,SAAb,gB;UAAa,WAAb,UAAa,SAAb,O;UACI,WAA,Y,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,iBAAnB,CAAJ,C;;QACHB,OAAO,W;MACX,C;KAXA,C;0FAaA,yB;MAAA,+D;MAAA,uC;QAKW,kBAAa,gB;QA0pDJ,Q;QAahB,iD;UAAgB,cAAhB,e;UAlpDqB,U;UAAA,cARe,SAQf,CAkpDQ,OAlpDR,W;YAAcS,6B;;;QAR3D,OASO,W;MARX,C;KANA,C;8FAQA,yB;MAAA,oD;QAupDoB,Q;QAahB,iD;UAAgB,cAAhB,e;U

AlpDqB,U;UAAA,wBAkpDQ,OAlpDR,W;YAAaC,6B;;;QAC3D,OAAO,W;MACX,C;KAPA,C;;MAciB,Q;MAAb ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAAY,WAAl,UAAU,IAAV,CAAJ,C;;MACHB,OAAO,W;IAC X,C;;MAOiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAAY,WAAl,UAAU,IAAV,CAAJ,C;; MACHB,OAAO,W;IACX,C;;MAOiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAAY,WAAl, UAAU,IAAV,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAOiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SAAb, M;QACI,WAAY,WAAl,UAAU,IAAV,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAOiB,Q;MAAb,wBAAa,SAAb,g B;QAAa,WAAA,SAAb,M;QACI,WAAY,WAAl,UAAU,IAAV,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAOiB,Q; MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAAY,WAAl,UAAU,IAAV,CAAJ,C;;MACHB,OAAO, W;IACX,C;;MAOiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAAY,WAAl,UAAU,IAAV,C AAJ,C;;MACHB,OAAO,W;IACX,C;;MAOiB,Q;MAAb,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,WAAY ,WAAl,UAAU,IAAV,CAAJ,C;;MACHB,OAAO,W;IACX,C;kFAEA,yB;MAAA,oC;MAAA,gC;MAAA,oD;QAKi B,Q;QAAb,wBAAa,SAAb,gB;UAAa,WAAb,UAAa,SAAb,O;UACI,WAAY,WAAl,UAAU,iBAAV,CAAJ,C;;QAC hB,OAAO,W;MACX,C;KARA,C;;MAe4B,mB;QAAE,2C;MAAF,C;IAAA,C;;MAAxB,OAAO,qBAAiB,2BAAjB, C;IACX,C;;MAO4B,mB;QAAE,+C;MAAF,C;IAAA,C;;MAAxB,OAAO,qBAAiB,6BAAjB,C;IACX,C;;MAO4B, mB;QAAE,gD;MAAF,C;IAAA,C;;MAAxB,OAAO,qBAAiB,6BAAjB,C;IACX,C;;MAO4B,mB;QAAE,8C;MAAF, C;IAAA,C;;MAAxB,OAAO,qBAAiB,6BAAjB,C;IACX,C;;MAO4B,mB;QAAE,+C;MAAF,C;IAAA,C;;MAAxB, OAAO,qBAAiB,6BAAjB,C;IACX,C;;MAO4B,mB;QAAE,gD;MAAF,C;IAAA,C;;MAAxB,OAAO,qBAAiB,6BA AjB,C;IACX,C;;MAO4B,mB;QAAE,iD;MAAF,C;IAAA,C;;MAAxB,OAAO,qBAAiB,6BAAjB,C;IACX,C;;MAO 4B,mB;QAAE,kD;MAAF,C;IAAA,C;;MAAxB,OAAO,qBAAiB,6BAAjB,C;IACX,C;;MAO4B,mB;QAAE,+C;MA AF,C;IAAA,C;;MAAxB,OAAO,qBAAiB,6BAAjB,C;IACX,C;;MAQI,OAA2B,SAAf,aAAL,SAAK,CAAE,C;IAC/ B,C;;MAQI,OAA2B,SAAf,eAAL,SAAK,CAAE,C;IAC/B,C;;MAQI,OAA2B,SAAf,eAAL,SAAK,CAAE,C;IAC/B, C;;MAQI,OAA2B,SAAf,eAAL,SAAK,CAAE,C;IAC/B,C;;MAQI,OAA2B,SAAf,eAAL,SAAK,CAAE,C;IAC/B,C;; MAQI,OAA2B,SAAf,eAAL,SAAK,CAAE,C;IAC/B,C;;MAQI,OAA2B,SAAf,eAAL,SAAK,CAAE,C;IAC/B,C;;M AQI,OAA2B,SAAf,eAAL,SAAK,CAAE,C;IAC/B,C;;MAQI,OAA2B,SAAf,eAAL,SAAK,CAAE,C;IAC/B,C;0FAE A,yB;MAAA,2D;MAAA,+D;MAAA,sC;QASc,Q;QAFV,UAAU,c;QACV,WAAW,gB;QACX,wBAAU,SAAV,gB; UAAU,QAAA,SAAV,M;UACI,UAAU,SAAS,CAAT,C;UACV,IAAI,GAAl,WAAl,GAAl,CAAR,C;YACI,IAAK, WAAl,CAAJ,C;;QAEb,OAAO,I;MACX,C;KafA,C;4FAiBA,yB;MAAA,2D;MAAA,+D;MAAA,sC;QASc,Q;QAF V,UAAU,c;QACV,WAAW,gB;QACX,wBAAU,SAAV,gB;UAAU,QAAA,SAAV,M;UACI,UAAU,SAAS,CAAT, C;UACV,IAAI,GAAl,WAAl,GAAl,CAAR,C;YACI,IAAK,WAAl,CAAJ,C;;QAEb,OAAO,I;MACX,C;KafA,C;4F AiBA,yB;MAAA,2D;MAAA,+D;MAAA,sC;QASc,Q;QAFV,UAAU,c;QACV,WAAW,gB;QACX,wBAAU,SAAV ,gB;UAAU,QAAA,SAAV,M;UACI,UAAU,SAAS,CAAT,C;UACV,IAAI,GAAl,WAAl,GAAl,CAAR,C;YACI,IA AK,WAAl,CAAJ,C;;QAEb,OAAO,I;MACX,C;KafA,C;4FAiBA,yB;MAAA,2D;MAAA,+D;MAAA,sC;QASc,Q; QAFV,UAAU,c;QACV,WAAW,gB;QACX,wBAAU,SAAV,gB;UAAU,QAAA,SAAV,M;UACI,UAAU,SAAS,CA AT,C;UACV,IAAI,GAAl,WAAl,GAAl,CAAR,C;YACI,IAAK,WAAl,CAAJ,C;;QAEb,OAAO,I;MACX,C;KafA, C;4FAiBA,yB;MAAA,2D;MAAA,+D;MAAA,sC;QASc,Q;QAFV,UAAU,c;QACV,WAAW,gB;QACX,wBAAU,S AAV,gB;UAAU,QAAA,SAAV,M;UACI,UAAU,SAAS,CAAT,C;UACV,IAAI,GAAl,WAAl,GAAl,CAAR,C;YA CI,IAAK,WAAl,CAAJ,C;;QAEb,OAAO,I;MACX,C;KafA,C;4FAiBA,yB;MAAA,2D;MAAA,+D;MAAA,sC;QA Sc,Q;QAFV,UAAU,c;QACV,WAAW,gB;QACX,wBAAU,SAAV,gB;UAAU,QAAA,SAAV,M;UACI,UAAU,SAAS,CA AT,C;UACV,IAAI,GAAl,WAAl,GAAl,CAAR,C;YACI,IAAK,WAAl,CAAJ,C;;QAEb,OAAO,I;MACX,C;K AfA,C;4FAiBA,yB;MAAA,2D;MAAA,+D;MAAA,sC;QASc,Q;QAFV,UAAU,c;QACV,WAAW,gB;QACX,wBA AU,SAAV,gB;UAAU,QAAA,SAAV,M;UACI,UAAU,SAAS,CAAT,C;UACV,IAAI,GAAl,WAAl,GAAl,CAAR,C ;YACI,IAAK,WAAl,CAAJ,C;;QAEb,OAAO,I;MACX,C;KafA,C;4FAiBA,yB;MAAA,2D;MAAA,+D;MAAA,sC; QASc,Q;QAFV,UAAU,c;QACV,WAAW,gB;QACX,wBAAU,SAAV,gB;UAAU,QAAA,SAAV,M;UACI,UAAU,SA AS,CAAT,C;UACV,IAAI,GAAl,WAAl,GAAl,CAAR,C;YACI,IAAK,WAAl,CAAJ,C;;QAEb,OAAO,I;MACX,C;K AfA,C;4FAiBA,yB;MAAA,2D;MAAA,+D;MAAA,sC;QASc,Q;QAFV,UAAU,c;QACV,WAAW,gB;QACX,wBA AU,SAAV,gB;UAAU,QAAA,SAAV,M;UACI,UAAU,SAAS,CAAT,C;UACV,IAAI,GAAl,WAAl,GAAl,CAAR,C ;YACI,IAAK,WAAl,CAAJ,C;;QAEb,OAAO,I;MACX,C;KafA,C;4FAiBA,yB;MAAA,2D;MAAA,+D;MAAA,sC; QA Sc,Q;QAFV,UAAU,c;QACV,WAAW,gB;QACX,wBAAU,SAAV,gB;UAAU,QAAV,UAAU,SAAV,O;UACI,UAAU,SAAS,cAAT,C;UACV,I AAI,GAAl,WAAl,GAAl,CAAR,C;YACI,IAAK,WAAl,cAAJ,C;;QAEb,OAAO,I;MACX,C;KafA,C;;MAuBI,UA Ae,aAAL,SAAK,C;MACX,YAAJ,GAAl,EAAU,KAAV,C;MACJ,OAAO,G;IACX,C;;MAQI,UAAe,eAAL,SAAK,

AAU,WAAV,EAAuB,oBAAvB,C;;QACpC,OAAO,W;MACX,C;KAPA,C;;MAkBoB,UAA8B,M;MAF9C,YAA Y,
C;MACZ,kBAAkB,O;MACIB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,cAAc,WAAU,cAAV,EA
AU,sBAAV,WAAmB,WAAAnB,EAAGC,OAAhC,C;;MACpC,OAAO,W;IACX,C;;MAWoB,UAA8B,M;MAF9C,Y
AA Y,C;MACZ,kBAAkB,O;MACIB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,cAAc,WAAU,cAA
V,EAAU,sBAAV,WAAmB,WAAAnB,EAAGC,OAAhC,C;;MACpC,OAAO,W;IACX,C;;MAWoB,UAA8B,M;MAF
9C,YAA Y,C;MACZ,kBAAkB,O;MACIB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,cAAc,WAAU,
cAAV,EAAU,sBAAV,WAAmB,WAAAnB,EAAGC,OAAhC,C;;MACpC,OAAO,W;IACX,C;;MAWoB,UAA8B,M;
MAF9C,YAA Y,C;MACZ,kBAAkB,O;MACIB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,cAAc,W
AAU,cAAV,EAAU,sBAAV,WAAmB,WAAAnB,EAAGC,OAAhC,C;;MACpC,OAAO,W;IACX,C;;MAWoB,UAA8
B,M;MAF9C,YAA Y,C;MACZ,kBAAkB,O;MACIB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,cA
Ac,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAAnB,EAAGC,OAAhC,C;;MACpC,OAAO,W;IACX,C;;MAWoB,
UAA8B,M;MAF9C,YAA Y,C;MACZ,kBAAkB,O;MACIB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAs
B,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAAnB,EAAGC,OAAhC,C;;MACpC,OAAO,W;IACX,C;;MA
WoB,UAA8B,M;MAF9C,YAA Y,C;MACZ,kBAAkB,O;MACIB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;
QAAsB,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAAnB,EAAGC,OAAhC,C;;MACpC,OAAO,W;IACX,C;
;MAWoB,UAA8B,M;MAF9C,YAA Y,C;MACZ,kBAAkB,O;MACIB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAh
B,M;QAAsB,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAAnB,EAAGC,OAAhC,C;;MACpC,OAAO,W;IA
CX,C;8FAEA,yB;MAAA,oC;MAAA,gC;MAAA,gD;QASoB,UAA8B,M;QAF9C,YAA Y,C;QACZ,kBAAkB,O;Q
ACIB,wBAAgB,SAAhB,gB;UAGB,cAAhB,UAGB,SAAhB,O;UAAsB,cAAc,WAAU,cAAV,EAAU,sBAAV,W
AAmB,WAAAnB,EAAGC,oBAAhC,C;;QACpC,OAAO,W;MACX,C;KAXA,C;wFAaA,yB;MAAA,8D;MAAA,gD;
QAOoC,Q;QAHhC,YAA Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,Y
AAJ,EAAl,oBAAJ,OAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAVA,C;0FAYA,yB;MAAA,8D;M
AAA,gD;QAOoC,Q;QAHhC,YAA Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU
,UAAI,YAAJ,EAAl,oBAAJ,OAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAVA,C;0FAYA,yB;MAA
A,8D;MAAA,gD;QAOoC,Q;QAHhC,YAA Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAA
c,UAAU,UAAI,YAAJ,EAAl,oBAAJ,OAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAVA,C;0FAYA,
yB;MAAA,8D;MAAA,gD;QAOoC,Q;QAHhC,YAA Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;U
ACI,cAAc,UAAU,UAAI,YAAJ,EAAl,oBAAJ,OAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAVA,C
;0FAYA,yB;MAAA,8D;MAAA,gD;QAOoC,Q;QAHhC,YAA Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CA
AhB,C;UACI,cAAc,UAAU,UAAI,YAAJ,EAAl,oBAAJ,OAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;
KAVA,C;0FAYA,yB;MAAA,8D;MAAA,gD;QAOoC,Q;QAHhC,YAA Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,S
AAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,YAAJ,EAAl,oBAAJ,OAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;
MACX,C;KAVA,C;0FAYA,yB;MAAA,8D;MAAA,gD;QAOoC,Q;QAHhC,YAA Y,wB;QACZ,kBAAkB,O;QACI
B,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,YAAJ,EAAl,oBAAJ,OAAV,EAawB,WAAxB,C;;QAEIB,
OAAO,W;MACX,C;KAVA,C;0FAYA,yB;MAAA,8D;MAAA,gD;QAOoC,Q;QAHhC,YAA Y,wB;QACZ,kBAAk
B,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,YAAJ,EAAl,oBAAJ,OAAV,EAawB,WAAxB,
C;;QAEIB,OAAO,W;MACX,C;KAVA,C;0FAYA,yB;MAAA,8D;MAAA,oC;MAAA,gD;QAOoC,Q;QAHhC,YAA
Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,sBAAl,YAAJ,EAAl,oBAAJ,QAA
V,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAVA,C;sGAYA,yB;MAAA,8D;MAAA,gD;QAOI,YAA Y,w
B;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAlB,UAAI,KAAJ,CAAjB,
EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAdA,C;wGAgBA,yB;MAAA,8D;MAAA,gD;QAOI,Y
AA Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAlB,UAAI,KAAJ,
CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAdA,C;wGAgBA,yB;MAAA,8D;MAAA,gD;
QAOI,YAA Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAlB,UAAI
,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAdA,C;wGAgBA,yB;MAAA,8D;MA
AA,gD;QAOI,YAA Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAl
B,UAAI,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAdA,C;wGAgBA,yB;MAAA,
8D;MAAA,gD;QAOI,YAA Y,wB;QACZ,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAA

V,EAAiB,UAAI,KAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAdA,C;wGAgBA,yB;
MAAA,8D;MAAA,gD;QAOI,YAAY,wB;QACZ,kBAaKB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAA
U,KAAV,EAAiB,UAAI,KAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAdA,C;wGA
gBA,yB;MAAA,8D;MAAA,gD;QAOI,YAAY,wB;QACZ,kBAaKB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cA
Ac,UAAU,KAAV,EAAiB,UAAI,KAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAdA,
C;wGAgBA,yB;MAAA,8D;MAAA,gD;QAOI,YAAY,wB;QACZ,kBAaKB,O;QACIB,OAAO,SAAS,CAAhB,C;U
ACI,cAAc,UAAU,KAAV,EAAiB,UAAI,KAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C
;KAdA,C;wGAgBA,yB;MAAA,8D;MAAA,oC;MAAA,gD;QAOI,YAAY,wB;QACZ,kBAaKB,O;QACIB,OAAO,
SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,sBAAL,KAJ,EAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,
OAAO,W;MACX,C;KAdA,C;;MAoBoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,OA
AO,OAAP,C;;IAC1B,C;;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,OAAO,O
AAP,C;;IAC1B,C;;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,OAAO,OAAP,
C;;IAC1B,C;;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,OAAO,OAAP,C;;IA
C1B,C;;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,OAAO,OAAP,C;;IAC1B,C;
;MAMoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,OAAO,OAAP,C;;IAC1B,C;;MA
MoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,OAAO,OAAP,C;;IAC1B,C;;MAMoB,
Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,OAAO,OAAP,C;;IAC1B,C;;sFAEA,yB;MA
AA,oC;MAAA,gC;MAAA,oC;QAIoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UAA
sB,OAAO,oBAAP,C;;MAC1B,C;KALA,C;;MAciB,UAAa,M;MAD1B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa
,WAAA,SAAb,M;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;IACvB,C;;MASiB,UAAa,M;MAD1
B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IA
AhB,C;;IACvB,C;;MASiB,UAAa,M;MAD1B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QA
AmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;IACvB,C;;MASiB,UAAa,M;MAD1B,YAAY,C;MACZ,w
BAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;IACvB,C;;M
ASiB,UAAa,M;MAD1B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAmB,QAAO,cAAP,EA
AO,sBAAP,WAAgB,IAAhB,C;;IACvB,C;;MASiB,UAAa,M;MAD1B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,
WAAA,SAAb,M;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;IACvB,C;;MASiB,UAAa,M;MAD1B
,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAAmB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IA
AhB,C;;IACvB,C;;MASiB,UAAa,M;MAD1B,YAAY,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QAA
mB,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;IACvB,C;oGAEa,yB;MAAA,oC;MAAA,gC;MAAA,oC;Q
AOiB,UAAa,M;QAD1B,YAAY,C;QACZ,wBAAa,SAAb,gB;UAAa,WAAb,UAAa,SAAb,O;UAAmB,QAAO,cAA
P,EAAO,sBAAP,WAAgB,iBAAhB,C;;MACvB,C;KARA,C;;MAoBiB,Q;MAHb,IA97IO,qBAAQ,CA87If,C;QAAe
,OAAO,I;MActB,UAAU,UAAK,CAAL,C;MACV,IAAQ,QAAJ,GAAL,CAAR,C;QAAiB,OAAO,G;MACX,+B;MA
AAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAM,QAAF,CAAE,CAAN,C;UAAe,OAAO,C;QA
CtB,IAAL,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAYiB,Q;MAHb,IAh9IO,qBAAQ,C
Ag9If,C;QAAe,OAAO,I;MActB,UAAU,UAAK,CAAL,C;MACV,IAAQ,QAAJ,GAAL,CAAR,C;QAAiB,OAAO,G
;MACX,+B;MAAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAM,QAAF,CAAE,CAAN,C;UAAe
,OAAO,C;QActB,IAAL,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IA/9I
O,qBAAQ,CA+9If,C;QAAe,OAAO,I;MActB,UAAU,UAAK,CAAL,C;MACG,+B;MAAAb,aAAU,CAAV,iB;QACI
,QAAQ,UAAK,CAAL,C;QACR,IAAL,sBAAM,CAAN,KAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;M
AQiB,Q;MAFb,IAp+IO,qBAAQ,CAo+If,C;QAAe,OAAO,I;MActB,UAAU,UAAK,CAAL,C;MACG,iC;MAAAb,a
AAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAL,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO
,G;IACX,C;;MAQiB,Q;MAFb,IAz+IO,qBAAQ,CAy+If,C;QAAe,OAAO,I;MActB,UAAU,UAAK,CAAL,C;MAC
G,iC;MAAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAL,MAAM,CAAV,C;UAAa,MAAM,C;;M
AEvB,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IA9+IO,qBAAQ,CA8+If,C;QAAe,OAAO,I;MActB,UAAU,UAAK,C
AAL,C;MACG,iC;MAAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAL,MAAM,CAAV,C;UAAa,
MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IA/IO,qBAAQ,CAM/If,C;QAAe,OAAO,I;MActB,UA
AU,UAAK,CAAL,C;MACG,iC;MAAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAL,oBAAM,C

AAN,KAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAWiB,Q;MAHb,IA1/IO,qBAAQ,CA0/If,C;QAAe ,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACV,IAAQ,QAAJ,GAAL,CAAR,C;QAAiB,OAAO,G;MACX,iC;MA Ab,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAM,QAAF,CAAE,CAAN,C;UAAe,OAAO,C;QACt B,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAWiB,Q;MAHb,IAngJO,qBAAQ,CA mgJf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACV,IAAQ,QAAJ,GAAL,CAAR,C;QAAiB,OAAO,G; MACX,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAM,QAAF,CAAE,CAAN,C;UAAe, OAAO,C;QACtB,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IAIgJ O,qBAAQ,CAkgJf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI, QAAQ,UAAK,CAAL,C;QACR,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;gFAEA,yB ;MAAA,8D;MAAA,sC;QAMI,IAjIJO,qBAAQ,CAiJf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gB AAqB,cAAL,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aA AU,CAAV,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW, CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA, 8D;MAAA,sC;QAMI,IA/IJO,qBAAQ,CA+IjF,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,c AAL,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CA AV,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX, KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;M AAA,sC;QAMI,IA7mJO,qBAAQ,CA6mJf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cA AL,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAA V,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX, KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;M AAA,sC;QAMI,IA3nJO,qBAAQ,CA2nJf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL, SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,O AAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAA A,sC;QAMI,IAzoJO,qBAAQ,CayoJf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAA K,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa, SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;Y ACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAAA,sC;Q AMI,IAvpJO,qBAAQ,CAupJf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C; QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAA b,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI, UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAAA,sC;QAMI,I ArqJO,qBAAQ,CAqqJf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACr B,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M; UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAA U,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAAA,sC;QAMI,IANrJ O,qBAAQ,CAMrJf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACrB,IA AI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI ,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C; YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAAA,oC;MAAA,sC;QAMI ,IAjsJO,qBAAQ,CAisJf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACr B,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M; UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAA U,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;;MA4BiB,Q;MAFb,IArxJO,qBAAQ,CAqxJf,C;Q AAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,+B;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL, C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAA O,G;IACX,C;;MAQiB,Q;MAFb,IA1xJO,qBAAQ,CA0xJf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MA

CG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IA/xJO,qBAAQ,CA+xJf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IApyJO,qBAAQ,CAoyJf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IAzyJO,qBAAQ,CAyyJf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IA9yJO,qBAAQ,CA8yJf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IANzJO,qBAAQ,CAmzJf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IAxzJO,qBAAQ,CAwzJf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IA7zJO,qBAAQ,CA6zJf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,gBAAR,EAAa,cAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAYiB,Q;MAHb,IA74JO,qBAAQ,CA64Jf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACV,IAAQ,QA AJ,GA AI,CAAR,C;QAAiB,OAAO,G;MACX,+B;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAM,QA AF,CAAE,CAAN,C;UAAe,OAAO,C;QACtB,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAYiB,Q;MAHb,IA/5JO,qBAAQ,CA+5Jf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACV,IAAQ,QA AJ,GA AI,CAAR,C;QAAiB,OAAO,G;MACX,+B;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAM,QA AF,CAAE,CAAN,C;UAAe,OAAO,C;QACtB,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IA96JO,qBAAQ,CA86Jf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,+B;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,sBAAM,CAAN,KA AJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IAN7JO,qBAAQ,CAm7Jf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IAx7JO,qBAAQ,CAw7Jf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IA77JO,qBAAQ,CA67Jf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IAI8JO,qBAAQ,CAk8Jf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,oBAAM,CAAN,KA AJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAWiB,Q;MAHb,IAz8JO,qBAAQ,CAy8Jf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACV,IAAQ,QA AJ,GA AI,CAAR,C;QAAiB,OAAO,G;MACX,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAM,QA AF,CAAE,CAAN,C;UAAe,OAAO,C;QACtB,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAWiB,Q;MAHb,IAI9JO,qBAAQ,CAk9Jf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;gFAEA,yB;MAAA,8D;MAAA,sC;QAMI,IAhiKO,qBAAQ,CagiKf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,

KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;M
AAA,sC;QAMI,IA9iKO,qBAAQ,CA8iKf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAA
L,SAAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,
OAAa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KA
AJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAA
A,sC;QAMI,IA5jKO,qBAAQ,CA4jKf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,S
AAK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OA
Aa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,
C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAAA,sC;
QAMI,IA1kKO,qBAAQ,CA0kKf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SA
AK,C;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OA
Aa,SAAb,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;Y
ACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAAA,sC;Q
AMI,IAxIKO,qBAAQ,CAwIKf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C
;QACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SA
Ab,M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YAC
I,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAAA,sC;QAMI
,IAtmKO,qBAAQ,CAsmKf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;Q
ACrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,
M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,U
AAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAAA,sC;QAMI,IA
pnKO,qBAAQ,CAonKf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACr
B,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;
UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAA
U,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAAA,sC;QAMI,IAloK
O,qBAAQ,CAkoKf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QACrB,IA
AI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI
,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;
YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;kFAsBA,yB;MAAA,8D;MAAA,oC;MAAA,sC;QAMI
,IAhpKO,qBAAQ,CAGpKf,C;UAAe,OAAO,I;QACtB,cAAc,UAAK,CAAL,C;QACd,gBAAqB,cAAL,SAAK,C;QA
CrB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,oBAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,
M;UACI,QAAQ,UAAK,CAAL,C;UACR,QAAQ,SAAS,cAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,U
AAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KApBA,C;;MA4BiB,Q;MAFb,IApuKO,qBAAQ,CAouK
f,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,+B;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,C
AAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,
OAAO,G;IACX,C;;MAQiB,Q;MAFb,IAzuKO,qBAAQ,CayuKf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,
C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,C
AAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IA9uKO,qBA
AQ,CA8uKf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAA
Q,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,
C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IANvKO,qBAAQ,CAMvKf,C;QAAe,OAAO,I;MACtB,UAAU,U
AAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,G
AAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,I
AxvKO,qBAAQ,CAwvKf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB
;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAA
oC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IA7vKO,qBAAQ,CA6vKf,C;QAAe,OAAO,I;MACt
B,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAA
W,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQi

B,Q;MAFb,IAIwKO,qBAAQ,CAkwKf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAA U,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CA AjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,Q;MAFb,IAvwKO,qBAAQ,CAuwKf,C;QAAe,O AAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,iC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QAC R,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IA CX,C;;MAQiB,Q;MAFb,IA5wKO,qBAAQ,CA4wKf,C;QAAe,OAAO,I;MACtB,UAAU,UAAK,CAAL,C;MACG,i C;MAAb,aAAU,CAAV,iB;QACI,QAAQ,UAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,gBAAR,EAAa,cAAb,CA AX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQI,OA31KO,qBAAQ,C;IA41KnB,C;; MAQI,OA51KO,qBAAQ,C;IA61KnB,C;;MAQI,OA71KO,qBAAQ,C;IA81KnB,C;;MAQI,OA91KO,qBAAQ,C;IA +1KnB,C;;MAQI,OA/1KO,qBAAQ,C;IAg2KnB,C;;MAQI,OAh2KO,qBAAQ,C;IAi2KnB,C;;MAQI,OAj2KO,qBA AQ,C;IAk2KnB,C;;MAQI,OAl2KO,qBAAQ,C;IAM2KnB,C;;MAQI,OAn2KO,qBAAQ,C;IAo2KnB,C;;MAQoB,Q ;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO ,K;;MACrD,OAAO,I;IACX,C;;MAQoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAA I,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,K;;MACrD,OAAO,I;IACX,C;;MAQoB,Q;MAAhB,wBAAgB,SAAhB,g B;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,K;;MACrD,OAAO,I;IACX,C ;MAQoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UA AwB,OAAO,K;;MACrD,OAAO,I;IACX,C;;MAQoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M; QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,K;;MACrD,OAAO,I;IACX,C;;MAQoB,Q;MAAhB,wBAA gB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,K;;MACrD,OA AO,I;IACX,C;;MAQoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAAsB,IAAI,UAAU,OAAV ,CAAJ,C;UAAwB,OAAO,K;;MACrD,OAAO,I;IACX,C;;MAQoB,Q;MAAhB,wBAAgB,SAAhB,gB;QAAgB,cAA A,SAAhB,M;QAAsB,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,K;;MACrD,OAAO,I;IACX,C;iFAEA,yB;MA AA,oC;MAAA,gC;MAAA,uC;QAMoB,Q;QAAhB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UA AsB,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,OAAO,K;;QACrD,OAAO,I;MACX,C;KARA,C;kFAUA,yB;MAAA, 4F;MAAA,8D;MAAA,uC;QAOqB,Q;QAHjB,IApgLO,qBAAQ,CAogLf,C;UACI,MAAM,mCAA8B,+BAA9B,C; QACV,kBAaqB,UAAK,CAAL,C;QACJ,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAuB,UAA K,KAAL,CAAvB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;oFAaA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAOq B,Q;QAHjB,IAzgLO,qBAAQ,CAygLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAakB,UAAK,CAAL,C; QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAuB,UAAK,KAAL,CAAvB,C;;QAEIB,OA AO,W;MACX,C;KAXA,C;oFAaA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAOqB,Q;QAHjB,IA9gLO,qBAAQ,CA 8gLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAakB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd, yB;UACI,cAAc,UAAU,WAAV,EAAuB,UAAK,KAAL,CAAvB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;oFAaA, yB;MAAA,4F;MAAA,8D;MAAA,uC;QAOqB,Q;QAHjB,IAnhLO,qBAAQ,CAmhLf,C;UACI,MAAM,mCAA8B,+ BAA9B,C;QACV,kBAakB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,E AAuB,UAAK,KAAL,CAAvB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;oFAaA,yB;MAAA,4F;MAAA,8D;MAA A,uC;QAOqB,Q;QAHjB,IAxhLO,qBAAQ,CAwhLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAakB,UA AK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAuB,UAAK,KAAL,CAAvB,C ;QAEIB,OAAO,W;MACX,C;KAXA,C;oFAaA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAOqB,Q;QAHjB,IA7hLO ,qBAAQ,CA6hLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAakB,UAAK,CAAL,C;QACD,+B;QAAjB,i BAAC,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAuB,UAAK,KAAL,CAAvB,C;;QAEIB,OAAO,W;MACX,C;KA XA,C;oFAaA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAOqB,Q;QAHjB,IALiLO,qBAAQ,CAkiLf,C;UACI,MAAM, mCAA8B,+BAA9B,C;QACV,kBAakB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAA U,WAAV,EAAuB,UAAK,KAAL,CAAvB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;oFAaA,yB;MAAA,4F;MAA A,8D;MAAA,uC;QAOqB,Q;QAHjB,IAviLO,qBAAQ,CAuiLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kB AAkB,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAuB,UAAK,KAAL, CAAvB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;oFAaA,yB;MAAA,4F;MAAA,8D;MAAA,oC;MAAA,gC;MAA A,uC;QAOqB,Q;QAHjB,IA5iLO,qBAAQ,CA4iLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAakB,UA AK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,wBAAV,EAAuB,sBAAK,KAAL,EAAvB,

E;;QAEIB,OAAO,W;MACX,C;KAXA,C;gGAaA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAUqB,Q;QAHjB,IA5nL O,qBAAQ,CA4nLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAaQb,UAAK,CAAL,C;QACJ,+B;QAAjB,i BAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAEIB,OAAO, W;MACX,C;KAdA,C;kGAgBA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAUqB,Q;QAHjB,IApOLO,qBAAQ,CAoo Lf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAaKb,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,y B;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAEIB,OAAO,W;MACX,C;K AdA,C;kGAgBA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAUqB,Q;QAHjB,IA5oLO,qBAAQ,CA4oLf,C;UACI,M AAM,mCAA8B,+BAA9B,C;QACV,kBAaKb,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc, UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;kGAgB A,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAUqB,Q;QAHjB,IApOLO,qBAAQ,CAopLf,C;UACI,MAAM,mCAA8B, +BAA9B,C;QACV,kBAaKb,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV, EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;kGAgBA,yB;MAAA,4F; MAAA,8D;MAAA,uC;QAUqB,Q;QAHjB,IA5pLO,qBAAQ,CA4pLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QA CV,kBAaKb,UAAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB, EAA8B,UAAK,KAAL,CAA9B,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;kGAgBA,yB;MAAA,4F;MAAA,8D;MA AA,uC;QAUqB,Q;QAHjB,IApqLO,qBAAQ,CAoqLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAaKb,U AAK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK, KAAL,CAA9B,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;kGAgBA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAUqB, Q;QAHjB,IA5qLO,qBAAQ,CA4qLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAaKb,UAAK,CAAL,C;Q ACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B, C;;QAEIB,OAAO,W;MACX,C;KAdA,C;kGAgBA,yB;MAAA,4F;MAAA,8D;MAAA,uC;QAUqB,Q;QAHjB,IApr LO,qBAAQ,CAorLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAaKb,UAAK,CAAL,C;QACD,+B;QAAj B,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,UAAK,KAAL,CAA9B,C;;QAEIB,OAA O,W;MACX,C;KAdA,C;kGAgBA,yB;MAAA,4F;MAAA,8D;MAAA,oC;MAAA,gC;MAAA,uC;QAUqB,Q;QAHj B,IA5rLO,qBAAQ,CA4rLf,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAaKb,UAAK,CAAL,C;QACD,+B; QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAAV,EAAiB,wBAAjB,EAA8B,sBAaK,KAAL,EAA9B,E;;QAE IB,OAAO,W;MACX,C;KAdA,C;4FAgBA,yB;MAAA,8D;MAAA,4F;MAAA,uC;QAM6B,UAE0,M;QAJhC,YAA Y,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAaQb,UAAI,YAAJ,EAAL,o BAAJ,O;QACrB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,cAAJ,EAAL,sBAAJ,SAAV,EAawB,WAAx B,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;8FAaA,yB;MAAA,8D;MAAA,4F;MAAA,uC;QAM0B,UAEU,M;QAJ hC,YAA Y,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAaKb,UAAI,YAA J,EAAL,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,cAAJ,EAAL,sBAAJ,SAAV,EAaw B,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;8FAaA,yB;MAAA,8D;MAAA,4F;MAAA,uC;QAM0B,UAE U,M;QAJhC,YAA Y,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAaKb,U AAI,YAAJ,EAAL,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,cAAJ,EAAL,sBAAJ,SA AV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;8FAaA,yB;MAAA,8D;MAAA,4F;MAAA,uC;QA M0B,UAEU,M;QAJhC,YAA Y,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,k BAaKb,UAAI,YAAJ,EAAL,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,UAAI,cAAJ,EAAL,s BAAJ,SAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;8FAaA,yB;MAAA,8D;MAAA,4F;MAA A,uC;QAM0B,UAEU,M;QAJhC,YAA Y,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+B AA9B,C;QACrB,kBAaKb,UAAI,YAAJ,EAAL,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,U AAI,cAAJ,EAAL,sBAAJ,SAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;8FAaA,yB;MAAA,8 D;MAAA,4F;MAAA,uC;QAM0B,UAEU,M;QAJhC,YAA Y,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mC AA8B,+BAA9B,C;QACrB,kBAaKb,UAAI,YAAJ,EAAL,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc, UAAU,UAAI,cAAJ,EAAL,sBAAJ,SAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;8FAaA,yB;

MAAA,8D;MAAA,4F;MAAA,uC;QAM0B,UAEU,M;QAJhC,YAAy,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,M
AAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,UAAI,YAAJ,EAAI,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;U
ACI,cAAc,UAAU,UAAI,cAAJ,EAAI,sBAAJ,SAAV,EAAwB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;
8FAaA,yB;MAAA,8D;MAAA,4F;MAAA,oC;MAAA,gC;MAAA,uC;QAM0B,UAEU,M;QAJhC,YAAy,wB;QAC
Z,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,UAAI,YAAJ,EAAI,oBAAJ,O;Q
ACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,oBAAU,sBAAI,cAAJ,EAAI,sBAAJ,UAAV,EAAwB,wBAAxB,E;;Q
AEIB,OAAO,W;MACX,C;KAXA,C;0GAaA,yB;MAAA,8D;MAAA,4F;MAAA,uC;QAS6B,Q;QAFzB,YAAy,wB;
QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAaqB,UAAI,YAAJ,EAAI,oBAAJ
,O;QACrB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,UAAI,KAAJ,CAAjB,EAA6B,WAA7B,C
;UACd,qB;;QAEJ,OAAO,W;MACX,C;KafA,C;4GaiBA,yB;MAAA,8D;MAAA,4F;MAAA,uC;QAS0B,Q;QAFtB
,YAAy,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,UAAI,YAAJ,E
AAI,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,UAAI,KAAJ,CAAjB,EAA6
B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KafA,C;4GaiBA,yB;MAAA,8D;MAAA,4F;MAAA,uC;QAS
0B,Q;QAFtB,YAAy,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,U
AAI,YAAJ,EAAI,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,UAAI,KAAJ,C
AAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KafA,C;4GaiBA,yB;MAAA,8D;MAAA,4F;MA
AA,uC;QAS0B,Q;QAFtB,YAAy,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB
,kBAakB,UAAI,YAAJ,EAAI,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,UA
AI,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KafA,C;4GaiBA,yB;MAAA,8D;M
AAA,4F;MAAA,uC;QAS0B,Q;QAFtB,YAAy,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA
9B,C;QACrB,kBAakB,UAAI,YAAJ,EAAI,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAA
V,EAAiB,UAAI,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KafA,C;4GaiBA,yB;
MAAA,8D;MAAA,4F;MAAA,uC;QAS0B,Q;QAFtB,YAAy,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mC
AA8B,+BAA9B,C;QACrB,kBAakB,UAAI,YAAJ,EAAI,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,
UAAU,KAAV,EAAiB,UAAI,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KafA,C;4
GaiBA,yB;MAAA,8D;MAAA,4F;MAAA,uC;QAS0B,Q;QAFtB,YAAy,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,
MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,UAAI,YAAJ,EAAI,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;
UACI,cAAc,UAAU,KAAV,EAAiB,UAAI,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,
C;KafA,C;4GaiBA,yB;MAAA,8D;MAAA,4F;MAAA,uC;QAS0B,Q;QAFtB,YAAy,wB;QACZ,IAAI,QAAQ,CA
AZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,UAAI,YAAJ,EAAI,oBAAJ,O;QACIB,OAAO,SAAS,
CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,UAAI,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAA
O,W;MACX,C;KafA,C;4GaiBA,yB;MAAA,8D;MAAA,4F;MAAA,oC;MAAA,gC;MAAA,uC;QAS0B,Q;QAFtB,
YAAy,wB;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,UAAI,YAAJ,E
AAI,oBAAJ,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,oBAAU,KAAV,EAAiB,sBAAI,KAAJ,CAAjB,EAA
6B,wBAA7B,E;UACd,qB;;QAEJ,OAAO,W;MACX,C;KafA,C;;MASBoB,Q;MADhB,UAAe,C;MACf,wBAAgB,S
AAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,YAAO,SAAS,OAAT,CAAP,I;;MAEJ,OAAO,G;IACX,C;;MAOoB,Q;
MADhB,UAAe,C;MACf,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,YAAO,SAAS,OAAT,CAAP,I;;
MAEJ,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAe,C;MACf,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;
QACI,YAAO,SAAS,OAAT,CAAP,I;;MAEJ,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAe,C;MACf,wBAAgB,S
AAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,YAAO,SAAS,OAAT,CAAP,I;;MAEJ,OAAO,G;IACX,C;;MAOoB,Q;
MADhB,UAAe,C;MACf,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,YAAO,SAAS,OAAT,CAAP,I;;
MAEJ,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAe,C;MACf,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;
QACI,YAAO,SAAS,OAAT,CAAP,I;;MAEJ,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAe,C;MACf,wBAAgB,S
AAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,YAAO,SAAS,OAAT,CAAP,I;;MAEJ,OAAO,G;IACX,C;;MAOoB,Q;
MADhB,UAAe,C;MACf,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,YAAO,SAAS,OAAT,CAAP,I;;
MAEJ,OAAO,G;IACX,C;kFAEA,yB;MAAA,oC;MAAA,gC;MAAA,sC;QAKoB,Q;QADhB,UAAe,C;QACf,wBA
AgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UACI,YAAO,SAAS,oBAAT,CAAP,I;;QAEJ,OAAO,G;MA
CX,C;KATA,C;;MAGBoB,Q;MADhB,UAAkB,G;MACIB,wBAAgB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QACI,

OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAkB,G;MACIB,wBAAgB,SAAhB,g
B;QAAgB,cAAA,SAAhB,M;QACI,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAA
kB,G;MACIB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;I
ACX,C;;MAOoB,Q;MADhB,UAAkB,G;MACIB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OAAO,S
AAS,OAAT,C;;MAEX,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAkB,G;MACIB,wBAAgB,SAAhB,gB;QAAgB
,cAAA,SAAhB,M;QACI,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAkB,G;MA
CIB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;IACX,C;;
MAOoB,Q;MADhB,UAAkB,G;MACIB,wBAAgB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OAAO,SAAS,OA
AT,C;;MAEX,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAkB,G;MACIB,wBAAgB,SAAhB,gB;QAAgB,cAAA,S
AAhB,M;QACI,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;IACX,C;8FAEA,yB;MAAA,oC;MAAA,gC;MAAA,sC
;QAKoB,Q;QADhB,UAAkB,G;QACIB,wBAAgB,SAAhB,gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UACI,OAAO,S
AAS,oBAAT,C;;QAEX,OAAO,G;MACX,C;KATA,C;;MAeoB,UAMT,M;MANP,wBAAgB,SAAhB,gB;QAAgB,c
AAA,SAAhB,M;QACI,IAAI,eAAJ,C;UACI,MAAM,gCAAYB,2BAAwB,SAAXB,MAAZB,C;;MAId,OAAO,0D;I
ACX,C;wFAEA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAQoB,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wB
AAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;
;YAEN,MAAO,WAAI,OAAJ,C;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;MACX,C;KAhBA,C;0FAkBA,yB
;MAAA,+D;MAAA,6B;MAAA,uC;QAQoB,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,SAAhB,gB;
UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,MAAO,W
AAI,OAAJ,C;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAA
A,6B;MAAA,uC;QAQoB,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAA,SA
AhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;QA
Gf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAAA,6B;MAAA,uC;Q
AQoB,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI
,UAAU,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;QAGf,OAAO,cAAK,K
AAL,EAAY,MAAZ,C;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAQoB,Q;QAFhB,Y
AAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CA
AJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,
C;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAQoB,Q;QAFhB,YAAY,gB;QACZ,a
AAa,gB;QACb,wBAAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,KA
AM,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;MACX,C;KA
hBA,C;0FAkBA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAQoB,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wB
AAgB,SAAhB,gB;UAAgB,cAAA,SAAhB,M;UACI,IAAI,UAAU,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;Y
AEN,MAAO,WAAI,OAAJ,C;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;MACX,C;KAhBA,C;0FAkBA,yB;MAAA,+D;MAA
A,oC;MAAA,gC;MAAA,6B;MAAA,uC;QAQoB,Q;QAFhB,YAAY,gB;QACZ,aAAa,gB;QACb,wBAAgB,SAAhB,
gB;UAAgB,cAAhB,UAAgB,SAAhB,O;UACI,IAAI,UAAU,oBAAV,CAAJ,C;YACI,KAAM,WAAI,oBAAJ,C;;Y
AEN,MAAO,WAAI,oBAAJ,C;;QAGf,OAAO,cAAK,KAAL,EAAY,MAAZ,C;MACX,C;KAhBA,C;;MAoHI,Wk
B5pZO,MAAK,KIB4pZK,gBkB5pZL,EIBikZD,KA2FkB,OkB5pZjB,C;MIB6pZZ,WAAW,iBAaA,IAAb,C;MACX,
aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WA9FqB,GA8FP,UAAK,CAAL,CA9FO,EAAhB,KA8FqB,CAAM
,CAAN,CA9FF,CA8FrB,C;;MA9FT,OAGGO,I;IAFX,C;;MA0GI,WkB5qZO,MAAK,KIB4qZK,gBkB5qZL,EIB2k
ZD,KAiGkB,OkB5qZjB,C;MIB6qZZ,WAAW,iBAaA,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IA
AK,WA9GqB,GAoGP,UAAK,CAAL,CApGO,EAAhB,KAoGqB,CAAM,CAAN,CApGF,CAoGrB,C;;MApGT,OA
sGO,I;IArGX,C;;MAGHI,WkB5rZO,MAAK,KIB4rZK,gBkB5rZL,EIBqlZD,KAuGkB,OkB5rZjB,C;MIB6rZZ,WA
AW,iBAaA,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WA1GqB,GA0GP,UAAK,CAAL,CA1
GO,EAAhB,KA0GqB,CAAM,CAAN,CA1GF,CA0GrB,C;;MA1GT,OA4GO,I;IA3GX,C;;MAshI,WkB5sZO,MAA
K,KIB4sZK,gBkB5sZL,EIB+lZD,KA6GkB,OkB5sZjB,C;MIB6sZZ,WAAW,iBAaA,IAAb,C;MACX,aAAU,CAAV

,MAAkB,IAAIB,M;QACI,IAAK,WAhHqB,GAgHP,UAAK,CAAL,CAhHO,EAAnB,KAgHqB,CAAM,CAAN,CAhHF,CAgHrB,C;;MAHHT,OAKHO,I;IAjHX,C;;MA4HI,WkB5tZO,MAAK,KIB4tZK,gBk5tZL,ElBymZD,KAmHkB,OkB5tZjB,C;MIB6tZZ,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WAtHqB,GAsHP,UAAK,CAAL,CAtHO,EAAnB,KAsHqB,CAAM,CAAN,CAtHF,CAsHrB,C;;MATHT,OAWhO,I;IAvHX,C;;MAkII,WkB5uZO,MAAK,KIB4uZK,gBk5uZL,ElBmnZD,KAyHkB,OkB5uZjB,C;MIB6uZZ,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WA5HqB,GA4HP,UAAK,CAAL,CA5HO,EAAnB,KA4HqB,CAAM,CAAN,CA5HF,CA4HrB,C;;MA5HT,OA8HO,I;IA7HX,C;;MAwII,WkB5vZO,MAAK,KIB4vZK,gBk5vZL,ElB6nZD,KA+HkB,OkB5vZjB,C;MIB6vZZ,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WAlIqB,GakIP,UAAK,CAAL,CAlIO,EAAnB,KakIqB,CAAM,CAAN,CAlIF,CAkIrB,C;;MAIIT,OAoIO,I;IANIX,C;;MA8II,WkB5wZO,MAAK,KIB4wZK,gBk5wZL,ElBuoZD,KaqIkB,OkB5wZjB,C;MIB6wZZ,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WAxIqB,GAwIP,UAAK,CAAL,CAXIO,EAAnB,KAwIqB,CAAM,CAAN,CAXIF,CAwIrB,C;;MAxIT,OAoIO,I;IAzIX,C;;MAoJI,WkB5xZO,MAAK,KIB4xZK,gBk5xZL,ElBipZD,KA2IkB,OkB5xZjB,C;MIB6xZZ,WAAW,iBAAa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WA9IqB,GA8IP,sBAAK,CAAL,EA9IO,EAAnB,KA8IqB,CAAM,CAAN,CA9IF,CA8IrB,C;;MA9IT,OAgJO,I;IAIX,C;8EAEA,yB;MAAA,gE;Mk3pZA,iB;MIB2pZA,8C;QAQI,WkB5pZO,MAAK,KIB4pZK,gBk5pZL,ElB4pZW,KAAM,OkB5pZjB,C;QIB6pZZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;8EAgBA,yB;MAAA,gE;Mk3qZA,iB;MIB2qZA,8C;QAQI,WkB5qZO,MAAK,KIB4qZK,gBk5qZL,ElB4qZW,KAAM,OkB5qZjB,C;QIB6qZZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;Mk3rZA,iB;MIB2rZA,8C;QAQI,WkB5rZO,MAAK,KIB4rZK,gBk5rZL,ElB4rZW,KAAM,OkB5rZjB,C;QIB6rZZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;8EAgBA,yB;MAAA,gE;Mk3sZA,iB;MIB2sZA,8C;QAQI,WkB5sZO,MAAK,KIB4sZK,gBk5sZL,ElB4sZW,KAAM,OkB5sZjB,C;QIB6sZZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;Mk3tZA,iB;MIB2tZA,8C;QAQI,WkB5tZO,MAAK,KIB4tZK,gBk5tZL,ElB4tZW,KAAM,OkB5tZjB,C;QIB6tZZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;Mk3uZA,iB;MIB2uZA,8C;QAQI,WkB5uZO,MAAK,KIB4uZK,gBk5uZL,ElB4uZW,KAAM,OkB5uZjB,C;QIB6uZZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;Mk3vZA,iB;MIB2vZA,8C;QAQI,WkB5vZO,MAAK,KIB4vZK,gBk5vZL,ElB4vZW,KAAM,OkB5vZjB,C;QIB6vZZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;MAAA,oC;Mk3xZA,iB;MIB2xZA,8C;QAQI,WkB5xZO,MAAK,KIB4xZK,gBk5xZL,ElB4xZW,KAAM,OkB5xZjB,C;QIB6xZZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,sBAAK,CAAL,EA AV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;;MAqHoB,gB;MAHhB,gBAAgB,gB;MACHb,WAAW,iBk5ZJ,MAAK,KIBu4ZwB,wBA5FzB,KA4FyB,EAAwB,EAAXB,Ck5v4ZxZ,ElBu4ZqD,Sk5v4ZrD,CIBu4ZD,C;MACX,QAAQ,C;MACQ,OA9FL,KA8FK,W;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhGqB,GAgGP,UAAK,UAAAL,EA AK,kBAAL,SAhGO,EAAGI,OA hGJ,CAgGrB,C;;MAhGT,OakGO,I;IAjGX,C;;MA+GoB,gB;MAHhB,gBAAGB,gB;MACHb,WAAW,iBk5ZJ,MAAK,KIB5ZwB,wBApGzB,KAoGyB,EAAwB,EAAXB,Ck5Z5ZxZ,ElBy5ZqD,Sk5Z5ZrD,CIBy5ZD,C;MACX,QAAQ,C;MACQ,OA tGL,KAsGK,W;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxGqB,

GAwGP,UAAK,UAAL,EAAK,kBAAL,SAxGO,EAwGI,OAxGJ,CAwGrB,C;;MAxGT,OA0GO,I;IAzGX,C;;MAu
HoB,gB;MAHhB,gBAAgB,gB;MACHB,WAAW,iBkB36ZJ,MAAK,KIB26ZwB,wBA5GzB,KA4GyB,EAAwB,EA
AxB,CkB36ZxB,EIB26ZqD,SkB36ZrD,CIB26ZD,C;MACX,QAAQ,C;MACQ,OA9GL,KA8GK,W;MAAhB,OAA
gB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhHqB,GAgHP,UAAK,UA
AL,EAAK,kBAAL,SAhHO,EAgHI,OAhHJ,CAGhrB,C;;MAhHT,OAKHO,I;IAjHX,C;;MA+HoB,gB;MAHhB,gBA
AgB,gB;MACHB,WAAW,iBkB77ZJ,MAAK,KIB67ZwB,wBApHzB,KAoHyB,EAAwB,EAAxB,CkB77ZxB,EIB6
7ZqD,SkB77ZrD,CIB67ZD,C;MACX,QAAQ,C;MACQ,OAtHL,KAsHK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,y
B;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxHqB,GAwHP,UAAK,UAAL,EAAK,kBAAL,SAx
HO,EAwHI,OAxHJ,CAwHrB,C;;MAxHT,OA0HO,I;IAzHX,C;;MAuIoB,gB;MAHhB,gBAAgB,gB;MACHB,WAA
W,iBkB/8ZJ,MAAK,KIB+8ZwB,wBA5HzB,KA4HyB,EAAwB,EAAxB,CkB/8ZxB,EIB+8ZqD,SkB/8ZrD,CIB+8Z
D,C;MACX,QAAQ,C;MACQ,OA9HL,KA8HK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,S
AAT,C;UAAoB,K;QACpB,IAAK,WAhIqB,GAgIP,UAAK,UAAL,EAAK,kBAAL,SAhIO,EAgII,OAHIJ,CAGIrB,C
;;MAhIT,OAKIO,I;IAjIX,C;;MA+IoB,gB;MAHhB,gBAAgB,gB;MACHB,WAAW,iBkBj+ZJ,MAAK,KIBi+ZwB,w
BApIzB,KAoIyB,EAAwB,EAAxB,CkBj+ZxB,EIBi+ZqD,SkBj+ZrD,CIBi+ZD,C;MACX,QAAQ,C;MACQ,OAtIL,
KAsIK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAxI
qB,GAwIP,UAAK,UAAL,EAAK,kBAAL,SAxIO,EAwII,OAxIJ,CAwIrB,C;;MAxIT,OA0IO,I;IAzIX,C;;MAuJoB,
gB;MAHhB,gBAAgB,gB;MACHB,WAAW,iBkBn/ZJ,MAAK,KIBm/ZwB,wBA5IzB,KA4IyB,EAAwB,EAAxB,Ck
Bn/ZxB,EIBm/ZqD,SkBn/ZrD,CIBm/ZD,C;MACX,QAAQ,C;MACQ,OA9IL,KA8IK,W;MAAhB,OAAgB,cAAhB,
C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhJqB,GAgJP,UAAK,UAAL,EAAK,kB
AAL,SAhJO,EAgII,OAHIJ,CAGJrB,C;;MAhJT,OAKJO,I;IAjJX,C;;MA+JoB,gB;MAHhB,gBAAgB,gB;MACHB,W
AAW,iBkBrgaJ,MAAK,KIBqgawB,wBApJzB,KAoJyB,EAAwB,EAAxB,CkBrgaxB,EIBqgaqD,SkBrgarD,CIBqga
D,C;MACX,QAAQ,C;MACQ,OAtJL,KAsJK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SA
AT,C;UAAoB,K;QACpB,IAAK,WAxJqB,GAwJP,UAAK,UAAL,EAAK,kBAAL,SAxJO,EAwJI,OAxJJ,CAwJrB,
C;;MAxJT,OA0JO,I;IAzJX,C;;MAuKoB,gB;MAHhB,gBAAgB,gB;MACHB,WAAW,iBkBvhaJ,MAAK,KIBuhawB
,wBA5JzB,KA4JyB,EAAwB,EAAxB,CkBvhaxB,EIBuhaqD,SkBvharD,CIBuhaD,C;MACX,QAAQ,C;MACQ,OA9
JL,KA8JK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,W
AhKqB,GAgKP,sBAAK,UAAL,EAAK,kBAAL,UAhKO,EAgKI,OAHKJ,CAGKrB,C;;MAhKT,OAKKO,I;IAjKX,C
;+EAEA,yB;MAAA,kF;MAAA,gE;MkBr4ZA,iB;MIBq4ZA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QACHB,
WAAW,ekBv4ZJ,MAAK,KIBu4ZwB,wBAAN,KAAM,EAAwB,EAAxB,CkBv4ZxB,EIBu4ZqD,SkBv4ZrD,CIBu4
ZD,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,
K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAARB,CAAJ,C;;QAET,OAAO,I;
MACX,C;KAhBA,C;+EakBA,yB;MAAA,kF;MAAA,gE;MkBv5ZA,iB;MIBu5ZA,8C;QAWoB,UAEY,M;QAL5B,
gBAAgB,gB;QACHB,WAAW,ekBz5ZJ,MAAK,KIBy5ZwB,wBAAN,KAAM,EAAwB,EAAxB,CkBz5ZxB,EIBy5
ZqD,SkBz5ZrD,CIBy5ZD,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,KA
AK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAARB,CA
AJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;+EakBA,yB;MAAA,kF;MAAA,gE;MkBz6ZA,iB;MIBy6ZA,8C;QA
WoB,UAEY,M;QAL5B,gBAAgB,gB;QACHB,WAAW,ekB36ZJ,MAAK,KIB26ZwB,wBAAN,KAAM,EAAwB,E
AAxB,CkB36ZxB,EIB26ZqD,SkB36ZrD,CIB26ZD,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;UA
AgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAL,EAAK,kBAAL,SA
AV,EAAqB,OAARB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;+EakBA,yB;MAAA,kF;MAAA,gE;MkB37Z
A,iB;MIB27ZA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QACHB,WAAW,ekB77ZJ,MAAK,KIB67ZwB,wBA
AN,KAAM,EAAwB,EAAxB,CkB77ZxB,EIB67ZqD,SkB77ZrD,CIB67ZD,C;QACX,QAAQ,C;QACQ,uB;QAAhB,
OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UA
AL,EAAK,kBAAL,SAAV,EAAqB,OAARB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;+EakBA,yB;MAAA,k
F;MAAA,gE;MkB78ZA,iB;MIB68ZA,8C;QAWoB,UAEY,M;QAL5B,gBAAgB,gB;QACHB,WAAW,ekB/8ZJ,MA
AK,KIB+8ZwB,wBAAN,KAAM,EAAwB,EAAxB,CkB/8ZxB,EIB+8ZqD,SkB/8ZrD,CIB+8ZD,C;QACX,QAAQ,
C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WA
AI,UAAU,UAAK,UAAL,EAAK,kBAAL,SAAV,EAAqB,OAARB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;

+EakBA,yB;MAAA,kF;MAAA,gE;MkB/9ZA,iB;MIB+9ZA,8C;QAWoB,UAEY,M;QAL5B,gBAAGB,gB;QACHB, WAAW,ekBj+ZJ,MAAK,KIBi+ZwB,wBAAN,KAAM,EAawB,EAAXB,CkBJ+ZxB,ElBi+ZqD,SkBj+ZrD,CIBi+Z D,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K ;UACpB,IAAK,WAAI,UAAU,UAAK,UAAAL,EAAK,kBAAL,SAAV,EAAqB,OAARb,CAAJ,C;;QAET,OAAO,I;M ACX,C;KAhBA,C;+EakBA,yB;MAAA,kF;MAAA,gE;MkBj/ZA,iB;MIBi/ZA,8C;QAWoB,UAEY,M;QAL5B,gB AAGB,gB;QACHB,WAAW,ekBn/ZJ,MAAK,KIBm/ZwB,wBAAN,KAAM,EAawB,EAAXB,CkBn/ZxB,ElBm/ZqD ,SkBn/ZrD,CIBm/ZD,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,IAAI,KAAK, SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAAL,EAAK,kBAAL,SAAV,EAAqB,OAARb,CAAJ,C ;QAET,OAAO,I;MACX,C;KAhBA,C;+EakBA,yB;MAAA,kF;MAAA,gE;MkBngaA,iB;MIBmgaA,8C;QAWoB, UAEY,M;QAL5B,gBAAGB,gB;QACHB,WAAW,ekBrgaJ,MAAK,KIBqgawB,wBAAN,KAAM,EAawB,EAAXB,C kBrgaxB,ElBqgaqD,SkBrgarD,CIBqgaD,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;U ACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,UAAK,UAAAL,EAAK,kBAAL,SAAV,EAAq B,OAARb,CAAJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;+EakBA,yB;MAAA,kF;MAAA,gE;MAAA,oC;MkBrha A,iB;MIBqhaA,8C;QAWoB,UAEY,M;QAL5B,gBAAGB,gB;QACHB,WAAW,ekBvhaJ,MAAK,KIBuhawB,wBAA N,KAAM,EAawB,EAAXB,CkBvhaxB,ElBuhagD,SkBvharD,CIBuhaD,C;QACX,QAAQ,C;QACQ,uB;QAAhB,OA AgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,sBAAK,UAA L,EAAK,kBAAL,UAAV,EAAqB,OAARb,CAAJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;;MA0GI,WkBxnaO,MA AK,KIBwnaK,gBkBxnaL,ElBuiaD,KAiFkB,OkBxnajB,C;MIBynaZ,WAAW,iBAaA,IAAb,C;MACX,aAAU,CAAV ,MAAkB,IAAIB,M;QACI,IAAK,WApFqB,GAoFP,UAAK,CAAL,CAPFO,EAAnB,KAoFqB,CAAM,CAAN,CAPF F,CAoFrB,C;;MApFT,OAsFO,I;IARFX,C;;MAGGI,WkBxoaO,MAAK,KIBwoaK,gBkBxoaL,ElBijaD,KAuFkB,Ok BxojB,C;MIByoaZ,WAAW,iBAaA,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WA1FqB,GA 0FP,UAAK,CAAL,CA1FO,EAAnB,KA0FqB,CAAM,CAAN,CA1FF,CA0FrB,C;;MA1FT,OA4FO,I;IA3FX,C;;M AsGI,WkBxpaO,MAAK,KIBwpaK,gBkBxpaL,ElB2jaD,KA6FkB,OkBxpajB,C;MIBypaZ,WAAW,iBAaA,IAAb,C; MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WAhGqB,GAGGP,UAAK,CAAL,CAhGO,EAAnB,KAGGq B,CAAM,CAAN,CAhGF,CAGGrB,C;;MAhGT,OAKGO,I;IAjGX,C;;MA4GI,WkBxqaO,MAAK,KIBwqaK,gBkBx qaL,ElBqkaD,KAmGkB,OkBxqajB,C;MIByqaZ,WAAW,iBAaA,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M ;QACI,IAAK,WAtGqB,GAsGP,UAAK,CAAL,CAtGO,EAAnB,KAsGqB,CAAM,CAAN,CAtGF,CAsGrB,C;;MAt GT,OAwoGO,I;IAvGX,C;;MAkHI,WkBxraO,MAAK,KIBwraK,gBkBxraL,ElB+kaD,KAyGkB,OkBxrajB,C;MIByra Z,WAAW,iBAaA,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WA5GqB,GA4GP,UAAK,CAA L,CA5GO,EAAnB,KA4GqB,CAAM,CAAN,CA5GF,CA4GrB,C;;MA5GT,OA8GO,I;IA7GX,C;;MAwHI,WkBxsa O,MAAK,KIBwsaK,gBkBxsaL,ElBylaD,KA+GkB,OkBxsajB,C;MIBysaZ,WAAW,iBAaA,IAAb,C;MACX,aAAU, CAAV,MAAkB,IAAIB,M;QACI,IAAK,WAIHQB,GAKHP,UAAK,CAAL,CAIHO,EAAnB,KAKHQB,CAAM,CAA N,CAIHF,CAkHrB,C;;MAIHT,OAoHO,I;IANHX,C;;MA8HI,WkBxtaO,MAAK,KIBwtaK,gBkBxtaL,ElBmmaD,K AqHkB,OkBxtajB,C;MIBytaZ,WAAW,iBAaA,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,W AxHqB,GAWHP,UAAK,CAAL,CAXHO,EAAnB,KAWHqB,CAAM,CAAN,CAXHF,CAWHrB,C;;MAxHT,OA0HO ,I;IAzHX,C;;MAoII,WkBxuaO,MAAK,KIBwuaK,gBkBxuaL,ElB6maD,KA2HkB,OkBxujB,C;MIByuaZ,WAAW, iBAaA,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WA9HqB,GA8HP,sBAAK,CAAL,EA9HO ,EA8HE,YA9HrB,KA8HqB,CAAM,CAAN,EA9HF,CA8HrB,C;;MA9HT,OAGIO,I;IA/HX,C;+EAEA,yB;MAAA,g E;MkBvnaA,iB;MIBunaA,8C;QAQI,WkBxnaO,MAAK,KIBwnaK,gBkBxnaL,ElBwnaW,KAAM,OkBxnajB,C;QIB ynaZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL, CAAV,EAAMb,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;M kBvoaA,iB;MIBuoaA,8C;QAQI,WkBxoaO,MAAK,KIBwoaK,gBkBxoaL,ElBwoaW,KAAM,OkBxojB,C;QIByoa Z,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CA AV,EAAMb,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;MkB vpaA,iB;MIBupaA,8C;QAQI,WkBxpaO,MAAK,KIBwpaK,gBkBxpaL,ElBwpaW,KAAM,OkBxpajB,C;QIBypaZ, WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAA V,EAAMb,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;MkBvq aA,iB;MIBuqaA,8C;QAQI,WkBxqaO,MAAK,KIBwqaK,gBkBxqaL,ElBwqaW,KAAM,OkBxqajB,C;QIByqaZ,WA

AW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,E
AAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;MkBvraA,i
B;MIBuraA,8C;QAQI,WkBxraO,MAAK,KIBwraK,gBkBxraL,EIBwraW,KAAM,OkBxrajB,C;QIByraZ,WAAW,e
AAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAMb
,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;MkBvsaA,iB;MIB
usaA,8C;QAQI,WkBxsaO,MAAK,KIBwsaK,gBkBxsaL,EIBwsaW,KAAM,OkBxsajB,C;QIBysaZ,WAAW,eAAa,I
AAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAMb,MA
AM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;MkBvtaA,iB;MIButaA
,8C;QAQI,WkBxtaO,MAAK,KIBwtaK,gBkBxtaL,EIBwtaW,KAAM,OkBxtajB,C;QIBytaZ,WAAW,eAAa,IAAb,C;
QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,UAAK,CAAL,CAAV,EAAMb,MAAM,CA
AN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;+EAgBA,yB;MAAA,gE;MAAA,oC;MkBvuaA,iB;MIB
uuaA,8C;QAQI,WkBxuaO,MAAK,KIBwuaK,gBkBxuaL,EIBwuaW,KAAM,OkBxujB,C;QIByuaZ,WAAW,eAAa,
IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,sBAAK,CAAL,EA AV,EAAMb,kB
AAM,CAAN,EAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;;MAwB8D,yB;QAAA,YAA0B,I;MAAM,sB;
QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,y
B;QAAA,YAAoC,I;MAGvN,Q;MAFhB,MAAO,gBAAO,MAAP,C;MACP,YAAY,C;MACZ,wBAAGB,SAAhB,gB
;QAAgB,cAAA,SAAhB,M;QACI,IAAI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,
CAAR,IAAa,SAAS,KAA1B,C;UACW,gBAAP,MAAO,EAAC,OAAd,EA AuB,SAAvB,C;;UACJ,K;;MAEX,IAAI,
SAAS,CAAT,IAAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,MAAO,gBAAO,OAAP,C;MAC
P,OAAO,M;IACX,C;;MAUwD,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;
MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAuC,I;MAGpN,Q;MAFhB,MAA
O,gBAAO,MAAP,C;MACP,YAAY,C;MACZ,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,IAAI,iCAA
U,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,CAAR,IAAa,SAAS,KAA1B,C;UACI,IAAI,iB
AAJ,C;YACI,MAAO,gBAAO,UAAU,OAAV,CAAP,C;;YAEP,MAAO,gBAAO,OAAQ,WAAf,C;;UACR,K;;MA
EX,IAAI,SAAS,CAAT,IAAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,MAAO,gBAAO,OA
P,C;MACP,OAAO,M;IACX,C;;MAUyD,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,U
AAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAwC,I;MAGtN,Q;MAF
hB,MAAO,gBAAO,MAAP,C;MACP,YAAY,C;MACZ,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,IA
AI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,CAAR,IAAa,SAAS,KAA1B,C;UACI
,IAAI,iBAAJ,C;YACI,MAAO,gBAAO,UAAU,OAAV,CAAP,C;;YAEP,MAAO,gBAAO,OAAQ,WAAf,C;;UACR
,K;;MAEX,IAAI,SAAS,CAAT,IAAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,MAAO,gBAA
O,OAAP,C;MACP,OAAO,M;IACX,C;;MAUuD,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;Q
AAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAuC,I;MAG
pN,Q;MAFhB,MAAO,gBAAO,MAAP,C;MACP,YAAY,C;MACZ,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB
,M;QACI,IAAI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,CAAR,IAAa,SAAS,KA
A1B,C;UACI,IAAI,iBAAJ,C;YACI,MAAO,gBAAO,UAAU,OAAV,CAAP,C;;YAEP,MAAO,gBAAO,OAAQ,WA
Af,C;;UACR,K;;MAEX,IAAI,SAAS,CAAT,IAAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,
MAAO,gBAAO,OAAP,C;MACP,OAAO,M;IACX,C;;MAUyD,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E
;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAA
wC,I;MAGtN,Q;MAFhB,MAAO,gBAAO,MAAP,C;MACP,YAAY,C;MACZ,wBAAGB,SAAhB,gB;QAAgB,cAAA,
SAAhB,M;QACI,IAAI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,CAAR,IAAa,SA
AS,KAA1B,C;UACI,IAAI,iBAAJ,C;YACI,MAAO,gBAAO,UAAU,OAAV,CAAP,C;;YAEP,MAAO,gBAAO,OA

AQ,WAAf,C;;;UACR,K;;MAEX,IAAI,SAAS,CAAT,IAAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,MAAO,gBAAO,OAAP,C;MACP,OAAO,M;IACX,C;;MAU0D,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAyC,I;MAGxN,Q;MAFhB,MAAO,gBAAO,MAAP,C;MACP,YAAy,C;MACZ,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,IAAI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,CAAR,IAAa,SAAS,KAA1B,C;UACI,IAAI,iBAAJ,C;YACI,MAAO,gBAAO,UAAU,OAAV,CAAP,C;;YAEP,MAAO,gBAAO,OAAQ,WAAf,C;;;UACR,K;;MAEX,IAAI,SAAS,CAAT,IAAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,MAAO,gBAAO,OAAP,C;MACP,OAAO,M;IACX,C;;MAU2D,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAA0C,I;MAGIN,Q;MAFhB,MAAO,gBAAO,MAAP,C;MACP,YAAy,C;MACZ,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,IAAI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,CAAR,IAAa,SAAS,KAA1B,C;UACI,IAAI,iBAAJ,C;YACI,MAAO,gBAAO,UAAU,OAAV,CAAP,C;;YAEP,MAAO,gBAAO,OAAQ,WAAf,C;;;UACR,K;;MAEX,IAAI,SAAS,CAAT,IAAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,MAAO,gBAAO,OAAP,C;MACP,OAAO,M;IACX,C;;MAUwD,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAuC,I;MAGpN,Q;MAFhB,MAAO,gBAAO,MAAP,C;MACP,YAAy,C;MACZ,wBAAGB,SAAhB,gB;QAAgB,cAAhB,UAGB,SAAhB,O;QACI,IAAI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,CAAR,IAAa,SAAS,KAA1B,C;UACI,IAAI,iBAAJ,C;YACI,MAAO,gBAAO,UAAU,oBAAV,CAAP,C;;YAEP,MAAO,gBAAO,OAAP,C;;;UACR,K;;MAEX,IAAI,SAAS,CAAT,IAAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,MAAO,gBAAO,OAAP,C;MACP,OAAO,M;IACX,C;;MAUyC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAoC,I;MACIN,OAAO,kBAAO,sBAAP,EAAwB,SAAxB,EAAmC,MAAnC,EAA2C,OAA3C,EAAoD,KAApD,EAA2D,SAA3D,EAASe,SAAtE,CAAiF,W;IAC5F,C;;MAUkC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAuC,I;MAC9M,OAAO,oBAAO,sBAAP,EAAwB,SAAxB,EAAmC,MAAnC,EAA2C,OAA3C,EAAoD,KAApD,EAA2D,SAA3D,EAASe,SAAtE,CAAiF,W;IAC5F,C;;MAUmC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAwC,I;MACHN,OAAO,oBAAO,sBAAP,EAAwB,SAAxB,EAAmC,MAAnC,EAA2C,OAA3C,EAAoD,KAApD,EAA2D,SAA3D,EAASe,SAAtE,CAAiF,W;IAC5F,C;;MAUiC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAsC,I;MAC5M,OAAO,oBAAO,sBAAP,EAAwB,SAAxB,EAAmC,MAAnC,EAA2C,OAA3C,EAAoD,KAApD,EAA2D,SAA3D,EAASe,SAAtE,CAAiF,W;IAC5F,C;;MAUkC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAuC,I;MAC9M,OAAO,oBAAO,sBAAP,EAAwB,SAAxB,EAAmC,MAAnC,EAA2C,OAA3C,EAAoD,KAApD,EAA2D,SAA3D,EAASe,SAAtE,CAAiF,W;IAC5F,C;;MAUmC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAwC,I;MACHN,OAAO,oBAAO,sBAAP,EAAwB,SAAxB,EAAmC,MAAnC,EAA2C,OAA3C,EAAoD,KAApD,EAA2D,SAA3D,EAASe,SAAtE,CAAiF,W;IAC5F,C;;MAUoC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAyC,I;MACIN,OAAO,oBAAO,sBAAP,EAAwB,SAAxB,EAAmC,MAAnC,EAA2C,OAA3C,EAAoD,KAApD,EAA2D,SAA3D,EAASe,SAAtE,CAAiF,W;IAC5F,C;;MAUqC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAuC,I;MAC9M,OAAO,oBAAO,sBAAP,EAAwB,SAAxB,EAAmC,MAAnC,EAA2C,OAA3C,EAAoD,KAApD,EAA2D,SAA3D,EAASe,SAAtE,CAAiF,W;IAC5F,C;;MAOoB,mB;QAAE,OAAK,qBAAL,eAAK,C;MAAP,C;IAAA,C;;MADhB,IA54OO,qBAAQ,CA44Of,C;QAAe,OAAO,W;MACtB,kCAAGB,4BAAhB,C;IACJ,C;;MAOoB,mB;QAAE,OAA

AAO,O;;MAEX,OAAO,G;IACX,C;;MAQoB,Q;MADhB,UAAe,C;MACf,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,YAAO,O;;MAEX,OAAO,G;IACX,C;;MAQoB,Q;MADhB,UAAe,C;MACf,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,YAAO,OAAP,I;;MAEJ,OAAO,G;IACX,C;;MAQoB,Q;MADhB,Y;MACA,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,cAAO,OAAP,C;;MAEJ,OAAO,G;IACX,C;;MAQoB,Q;MADhB,UAAiB,G;MACjB,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OAAO,O;;MAEX,OAAO,G;IACX,C;;MAQoB,Q;MADhB,UAAkB,G;MACiB,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OAAO,O;;MAEX,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAe,C;MACf,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,YAAO,O;;MAEX,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAe,C;MACf,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,YAAO,OAAP,I;;MAEJ,OAAO,G;IACX,C;;MAOoB,Q;MADhB,Y;MACA,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,cAAO,OAAP,C;;MAEJ,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAiB,G;MACjB,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OAAO,O;;MAEX,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAkB,G;MACiB,wBAAGB,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QACI,OAAO,O;;MAEX,OAAO,G;IACX,C;;MY7qcuF,wC;IARvF,C;IAAA,0D;MAS2C,8B;IAAA,C;IAT3C,gF;;MOcI,OAAO,sBAAI,CAAJ,C;IACX,C;;MAOI,OAAO,sBAAI,CAAJ,C;IACX,C;;MAOI,OAAO,sBAAI,CAAJ,C;IACX,C;;MAOI,OAAO,sBAAI,CAAJ,C;IACX,C;;MAMI,IAAI,oCAAJ,C;QACI,OAAO,yBAAS,OAAT,C;MACX,OAAO,qBAAQ,OAAR,KAAoB,C;IAC/B,C;;MAUkC,qB;QAAE,MAAM,8BAA0B,iDAA8C,AA9C,MAA1B,C;MAAR,C;IAAA,C;;MAF9B,IAAI,8BAAJ,C;QACI,OAAO,sBAAI,KAAJ,C;MACX,OAAO,6BAAgB,KAAhB,EAAuB,uBAAvB,C;IACX,C;;MASI,OAAO,sBAAI,KAAJ,C;IACX,C;;MAgBqB,Q;MARjB,IAAI,8BAAJ,C;QACI,OAAsB,KAwJf,IAAS,CAAT,IAxJe,KAwJD,IAAS,iBAxJvB,SAwJuB,CAA3B,GAxJI,SAwJkC,aAxJnB,KAwJmB,CAAtC,GAxJ0B,YAwJ4B,CaxJnC,KAwJmC,C;;MAvJ7D,IAAI,QAAQ,CAAZ,C;QACI,OAAO,aAAa,KAAb,C;MACX,eAAe,oB;MACf,YAAY,C;MACZ,OAAO,QAAS,UAAhB,C;QACI,cAAc,QAAS,O;QACvB,IAAI,WAAS,YAAT,EAAS,oBAAT,OAAJ,C;UACI,OAAO,O;;MAEf,OAAO,aAAa,KAAb,C;IACX,C;sGAEA,yB;MAAA,8D;MAAA,iD;QAOI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAsC,sBAAI,KAAJ,CAAtC,GAA sD,aAAa,KAAb,C;MACjE,C;KARA,C;;MAwBqB,Q;MARjB,IAAI,8BAAJ,C;QACI,OAAY,YAAL,SAAK,EAAU,KAAV,C;MachB,IAAI,QAAQ,CAAZ,C;QACI,OAAO,I;MACX,eAAe,oB;MACf,YAAY,C;MACZ,OAAO,QAA S,UAAhB,C;QACI,cAAc,QAAS,O;QACvB,IAAI,WAAS,YAAT,EAAS,oBAAT,OAAJ,C;UACI,OAAO,O;;MAEf,OAAO,I;IACX,C;sGAEA,yB;MAAA,sD;MAAA,mC;QAOI,OAAY,UAAAL,SAAK,EAAU,KAAV,C;MachB,C;KARA,C;;MAeW,sB;;QAqFS,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UAAM,IArFH,SAqFO,CAAU,OA AV,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;QAC9C,qBAAO,I;;MAiFP,yB;IACJ,C;;MAuSoB,Q;MADhB,WAAe,I;MACC,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,IAjSc,SAiSV,CAAU,OA AV,CAAJ,C;UACI,OA AO,O;;MAISf,OAqSO,I;IApSX,C;;MAOW,qB;;QAoSP,eAAoB,+BAAa,cAAb,C;QACpB,OAAO,QAAS,cAAhB,C;UACI,cAAc,QAAS,W;UACvB,IAvSc,SAuSV,CAAU,OA AV,CAAJ,C;YAAwB,oBAAO,O;YAAP,sB;;QAE5B,oBAAO,I;;MAZSP,wB;IACJ,C;;MAQQ,kBADE,SACF,Q;QAAW,OAAY,SAAL,SAAK,C;;QAE nB,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAA d,C;UACI,MAAM,2BA AuB,sBA AvB,C;QACV,OAAO,QAAS,O;;IAG5B,C;;MAOI,IAAI,mBA AJ,C;QACI,MAAM,2BA AuB,gBA AvB,C;MACV,OAAO,sBA AK,CAAL,C;IACX,C;mFAEA,yB;MAA A,iE;MAAA,uC;QAKoB,Q;QAAA,2B;QA AhB,OAAGB,cAAhB,C;UAAgB,yB;UAAM,IAAI,UAAU,OA AV,CAA J,C;YAAwB,OAAO,O;;QACrD,MAAM,gCA AuB,wDA AvB,C;MACV,C;KAPA,C;;MAcQ,kBADE,SACF,Q;QAC I,IAAI,mBA AJ,C;UACI,OAAO,I;;UAEP,OAAO,sBA AK,CAAL,C;;QAGX,eAAe,oB;QACf,IAAI,CAAC,QAAS,U AA d,C;UACI,OAAO,I;QACX,OAAO,QAAS,O;;IAG5B,C;;MAMI,OA AW,mBA AJ,GAAe,IAAf,GAAyB,sBA AK,CAAL,C;IACpC,C;;MAMoB,Q;MAAA,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QAAM,IAAI,UAAU,OA AV,CAA J,C;YAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;0FAEA,yB;MAAA,8D;MAAA,iD;QAKI,OA AW,SAAS,C AAT,IAAc,SAAS,wBAA3B,GAAsC,sBA AI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;MACjE,C;KANA,C;;MAYI,OA AW,SAAS,CAAT,IAAc,SAAS,2BAA3B,GAAsC,sBA AI,KAAJ,CAAtC,GAAsD,I;IACjE,C;;MAQiB,Q;MAFb,IAAI,8BA AJ,C;QA AkB,OAAO,SAAK,eAAQ,OAAR,C;MAC9B,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QA Aa,sB;QACT,mBA AmB,KAA nB,C;QACA,IAAI,gBA AW,IAAX,CAAJ,C;UACI,OAAO,K;QACX,qB;;MAEJ,OA AO,E;IACX,C;;MAOI,OAAO,wBAAQ,OAAR,C;IACX,C;gGAEA,yB;MAAA,wE;MAAA,uC;QAKiB,Q;QADb,Y AAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,mBA AmB,KAA nB,C;UACA,IAAI,UAAU,IAAV,CA

AJ,C;YACI,OAAO,K;UACX,qB;;QAEJ,OAAO,E;MACX,C;KAZA,C;;MAMBiB,Q;MADb,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAI,UAAU,IAAV,CAAJ,C;UACI,OAAO,K;QACX,qB;;MAEJ,OAAO,E;IACX,C;8FAEA,yB;MAAA,wE;MAAA,uC;QAMiB,Q;QAFb,gBAAgB,E;QACbB,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,mBAAmB,KAAAnB,C;UACA,IAAI,UAAU,IAAV,CAAJ,C;YACI,YAAY,K;UAC hB,qB;;QAEJ,OAAO,S;MACX,C;KAbA,C;;MAMBI,eAAe,SAAK,sBAAa,cAAb,C;MACpB,OAAO,QAAS,cAAh B,C;QACI,IAAI,UAAU,QAAS,WAAAnB,CAAJ,C;UACI,OAAO,QAAS,Y;;;MAGxB,OAAO,E;IACX,C;;MAQQ,k BADE,SACF,Q;QAAW,OAAY,QAAL,SAAK,C;;QAEEnB,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAAd,C;UACI,M AAM,2BAAuB,sBAAvB,C;QACV,WAAW,QAAS,O;QACpB,OAAO,QAAS,UAAhB,C;UACI,OAAO,QAAS,O; QACpB,OAAO,I;;IAGnB,C;;MAOI,IAAI,mBAAJ,C;QACI,MAAM,2BAAuB,gBAAvB,C;MACV,OAAO,sBAAK, 2BAAL,C;IACX,C;iFAEA,yB;MAAA,iE;MAAA,gB;MAAA,8B;MAAA,uC;QAOoB,UAQT,M;QAVP,WAAe,I;Q ACf,YAAY,K;QACI,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,OA AV,CAAJ,C;YACI,OA AO,O;YACP,QAAQ,I;;QAGhB,IAAI,CAAC,KAAL,C;UAA Y,MAAM,gCAAuB,wDAAvB,C;QAEIB,OAAO,2E; MACX,C;KAhBA,C;iFAkBA,yB;MAAA,iE;MAAA,uC;QAKI,eAAe,SAAK,sBAAa,cAAb,C;QACpB,OAAO,QA AS,cAAhB,C;UACI,cAAc,QAAS,W;UACvB,IAAI,UAAU,OA AV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,MAAM,g CAAuB,kDAAvB,C;MACV,C;KAXA,C;;MAoBiB,Q;MAHb,IAAI,8BAAJ,C;QAAkB,OAAO,SAAK,mBAA Y,OA AZ,C;MAC9B,gBAAgB,E;MACHb,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,mBAAmB,KA AnB,C;QACA,IAAI,gBAAW,IAAX,CAAJ,C;UACI,YAAY,K;QACbB,qB;;MAEJ,OAAO,S;IACX,C;;MAOI,OAA O,4BAA Y,OAAZ,C;IACX,C;;MAOQ,kBADE,SACF,Q;QAAW,OA AW,mBAAJ,GAAe,IAAf,GAAyB,sBAAK,iB AAO,CAAP,IAAL,C;;QAEvC,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAAd,C;UACI,OAAO,I;QACX,WAAW,QA AS,O;QACpB,OAAO,QAAS,UAAhB,C;UACI,OAAO,QAAS,O;QACpB,OAAO,I;;IAGnB,C;;MAMI,OAAW,mB AAJ,GAAe,IAAf,GAAyB,sBAAK,iBAAO,CAAP,IAAL,C;IACpC,C;;MAOoB,Q;MADhB,WAAe,I;MACC,2B;M AAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,IAAI,UAAU,OA AV,CAAJ,C;UACI,OAAO,O;;MAGf,OAAO,I;IA CX,C;;MAMI,eAAe,SAAK,sBAAa,cAAb,C;MACpB,OAAO,QAAS,cAAhB,C;QACI,cAAc,QAAS,W;QACvB,IA AI,UAAU,OA AV,CAAJ,C;UAAwB,OAAO,O;;MAEnC,OAAO,I;IACX,C;qFAEA,yB;MAAA,mC;MAAA,gD;MA AA,4B;QAQI,OAAO,kBAAO,cAAP,C;MACX,C;KATA,C;;MAkBI,IAAI,mBAAJ,C;QACI,MAAM,2BAAuB,sB AA vB,C;MACV,OAAO,qBAAU,MAAO,iBAAQ,cAAR,CAAjB,C;IACX,C;;MAOQ,kBADE,SACF,Q;QAAW,OA AY,UAAAL,SAAK,C;;QAEEnB,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAAd,C;UACI,MAAM,2BAAuB,sBAAvB,C; QACV,aAAa,QAAS,O;QACtB,IAAI,QAAS,UAAAb,C;UACI,MAAM,gCAAyB,uCAAzB,C;QACV,OAAO,M;;IAG nB,C;;MAMiB,IAAN,I;MAAA,QAAM,cAAN,C;aACH,C;UAAK,MAAM,2BAAuB,gBAAvB,C;aACX,C;UAAK, 6BAAK,CAAL,C;UAAAL,K;gBACQ,MAAM,gCAAyB,iCAAzB,C;;MAHIB,W;IAKJ,C;qFAEA,yB;MAAA,kF;MA AA,iE;MAAA,gB;MAAA,8B;MAAA,uC;QAMoB,UAST,M;QAXP,aAAiB,I;QACjB,YAAY,K;QACI,2B;QAAhB, OAAGB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,OA AV,CAAJ,C;YACI,IAAI,KA AJ,C;cAAW,MAAM,8BAAy B,qDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;QAGhB,IAAI,CAAC,KAAL,C;UAA Y,MAAM,gCAAuB,wDAA vB,C;QAEIB,OAAO,6E;MACX,C;KAhBA,C;;MAuBQ,kBADE,SACF,Q;QAAW,OA AW,mBAAQ,CAAZ,GAAe, sBAAK,CAAL,CAAf,GAA4B,I;;QAEIC,eAAe,oB;QACf,IAAI,CAAC,QAAS,UAAAd,C;UACI,OAAO,I;QACX,aA Aa,QAAS,O;QACtB,IAAI,QAAS,UAAAb,C;UACI,OAAO,I;QACX,OAAO,M;;IAGnB,C;;MAMI,OAAW,mBAAQ, CAAZ,GAAe,sBAAK,CAAL,CAAf,GAA4B,I;IACvC,C;;MAQoB,Q;MAFhB,aAAiB,I;MACjB,YAAY,K;MACI,2 B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,IAAI,UAAU,OA AV,CAAJ,C;UACI,IAAI,KA AJ,C;YAAW,OA AO,I;UACIB,SAAS,O;UACT,QAAQ,I;;MAGhB,IAAI,CAAC,KAAL,C;QAAY,OAAO,I;MACnB,OAAO,M;IAC X,C;;MAsBsC,UAGT,MAHS,EAarB,M;MNpkBb,IAAI,EM2iBI,KAAK,CN3iBT,CAAJ,C;QACI,cM0iBc,sD;QNzi Bd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MM0iBV,IAAI,MAAK,CAAT,C;QAAY,OAAO,mB;MACnB,Q;MACA, IAAI,oCAAJ,C;QACI,iBAAiB,iBAAO,CAAP,I;QACjB,IAAI,cAAc,CAAIB,C;UACI,OAAO,W;QACX,IAAI,eAA c,CAAIB,C;UACI,OAAO,OAAO,kBAAP,C;QACX,OAAO,iBAAa,UAAAb,C;QACP,IAAI,8BAAJ,C;UACI,IAAI,s CAAJ,C;YAC0B,qB;YAAAtB,iBAAc,CAAd,wB;cACI,IAAK,WAAI,sBAAK,KAAL,CAAJ,C;;YAEI,wCAAa,CA Ab,C;YAAb,OAAa,gBAAb,C;cAAa,wB;cACT,IAAK,WAAI,IAAJ,C;;UAEb,OAAO,I;;QAI X,OAAO,gB;;MAE X,YAAY,C;MACC,6B;MAAb,OAAa,gBAAb,C;QAAa,0B;QACT,IAAI,SAAS,CAAb,C;UAAgB,IAAK,WAAI,M AAJ,C;;UAAe,qB;;MAEXC,OAAY,qBAAL,IAAK,C;IACHB,C;;MNxkBI,IAAI,EMklBI,KAAK,CNII BT,CAAJ,C; QACI,cMilBc,sD;QNhlBd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MMilBV,OAAO,kBAAGB,gBAAV,iBAAO,CAA

P,IAAU,EAAC,CAAd,CAAhB,C;IACX,C;kGAEA,yB;MAAA,4C;MAAA,qD;MAAA,uC;QAMI,IAAI,CAAC,mB
AAL,C;UACI,eAAe,+BAaA,cAAb,C;UACf,OAAO,QAAS,cAAhB,C;YACI,IAAI,CAAC,UAAU,QAAS,WAAAnB,
CAAL,C;cACI,OAAO,gBAAK,QAAS,YAAT,GAAuB,CAAvB,IAAL,C;;;QAIInB,OAAO,W;MACX,C;KAFa,C;0
FAiBA,yB;MAAA,+D;MAAA,uC;QAQiB,Q;QAFb,eAAe,K;QACf,WAAW,gB;QACE,2B;QAAb,OAAa,cAAb,C;
UAAa,sB;UACT,IAAI,QA AJ,C;YACI,IAAK,WAAI,IAAJ,C;eACJ,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,I
AAK,WAAI,IAAJ,C;YA CL,WAAW,I;;;QAEnB,OAAO,I;MACX,C;KAhBA,C;oFAkBA,yB;MAAA,+D;MAAA,u
C;QAIW,kBAAS,gB;QAyEA,Q;QAAA,2B;QA AhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAzEU,SAyEN,CAA
U,OAAV,CAAJ,C;YAAwB,WAA Y,WAAI,OAAJ,C;;QAzE1D,OA0EO,W;MAzEX,C;KALA,C;kGGOA,yB;MAA
A,+D;MAK5BA,wE;MAI5BA,uC;QAMW,kBAAgB,gB;QAm5BV,gB;QADb,YAA Y,C;QACC,2B;QAAb,OAAa,c
AAb,C;UAAa,sB;UAz4BT,IAVmC,SAU/B,CAY4BkB,oBAAmB,cAAAnB,EAAMB,sBAAnB,UAz4BIB,EAY4B+C,I
Az4B/C,CAAJ,C;YAA2C,sBAy4BQ,IAz4BR,C;;QAV/C,OAYO,W;MAXX,C;KAPA,C;sGASA,yB;MAy4BA,wE;
MAz4BA,oD;QAg5BiB,gB;QADb,YAA Y,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UAz4BT,IAAI,UAY4Bk
B,oBAAmB,cAAAnB,EAAMB,sBAAnB,UAz4BIB,EAY4B+C,IAz4B/C,CAAJ,C;YAA2C,sBAy4BQ,IAz4BR,C;;QA
E/C,OAAO,W;MACX,C;KAVA,C;wGAYA,yB;MAAA,+D;MAAA,sC;QAIW,kBAAmB,gB;QAOV,Q;QAAA,2B;
QA AhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,YAAJ,C;YAAkB,WAA Y,WAAI,OAAJ,C;;QAPpD,OAQO,
W;MAPX,C;KALA,C;;MAWOB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,YAAJ,C;UA
AkB,WAA Y,WAAI,OAAJ,C;;MACpD,OAAO,W;IACX,C;0FAEA,yB;MAAA,+D;MAAA,uC;QAIW,kBAAY,gB;
QAsBH,Q;QAAA,2B;QA AhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CatBS,SAsBR,CAAU,OAAV,CAAL,
C;YAAyB,WAA Y,WAAI,OAAJ,C;;QAtB3D,OAuBO,W;MatBX,C;KALA,C;;MAWI,OAAO,6BAAGB,gBAAhB,
C;IACX,C;;MAMOB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,eAAJ,C;UAAqB,WAA
Y,WAAI,OAAJ,C;;MACvD,OAAO,W;IACX,C;;MAMOB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;
QAAM,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAA Y,WAAI,OAAJ,C;;MAC3D,OAAO,W;IACX,C;;MA
MoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,WAA
Y,WAAI,OAAJ,C;;MAC1D,OAAO,W;IACX,C;;MAMI,IAAI,OAAQ,UAAZ,C;QAAuB,ObhqBe,W;;MaiqBtC,OA
A6D,SAAtD,SAAK,iBAAQ,OAAQ,MAAhB,EA AuB,OAAQ,aAAR,GAAuB,CAAvB,IAAvB,CAAiD,C;IACjE,C;;
MASkB,Q;MAHd,WAAmB,wBAAR,OAAQ,EA AwB,EA AxB,C;MACnB,IAAI,SAAQ,CAAZ,C;QA Ae,OAAO,W
;MACtB,WAAW,iBAaA,IAAb,C;MACG,yB;MAAd,OA Ac,cAAd,C;QA Ac,uB;QACV,IAAK,WAAI,sBAAI,KAA
J,CAAJ,C;;MAET,OAAO,I;IACX,C;;MAkBiB,UACL,M;MNHvBR,IAAI,EMuuBI,KAAK,CNvuBT,CAAJ,C;QAC
I,cMsuBc,sD;QNruBd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MMsuBV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;
MACnB,IAAI,oCAAJ,C;QACI,IAAI,KAAK,cAAT,C;UAAe,OAAO,mB;QACtB,IAAI,MAAK,CAAT,C;UAA Y,O
AAO,OAAO,mBAAP,C;;MAEvB,YAA Y,C;MACZ,WAAW,iBAaA,CAAb,C;MACE,2B;MAAb,OAAa,cAAb,C;Q
AAa,sB;QACT,IAAI,qDA AW,CAAF,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;MAET,OAA Y,qBAAL,IAAK,C;IA
ChB,C;;MAiBqC,IAGhB,I;MNzwBjB,IAAI,EM+vBI,KAAK,CN/vBT,CAAJ,C;QACI,cM8vBc,sD;QN7vBd,MAA
M,gCAAyB,OAAQ,WAAjC,C;;MM8vBV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,c;MACX,I
AAI,KAAK,IAAT,C;QA Ae,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,kBAAP,C;MACnB
,WAAW,iBAaA,CAAb,C;MACX,IAAI,sCAAJ,C;QACI,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;UACI,IA
AK,WAAI,sBAAK,KAAL,CAAJ,C;;;QAEI,sCAAa,OAAO,CAAP,IAAb,C;QAAb,OAAa,cAAb,C;UAAa,sB;UAC
T,IAAK,WAAI,IAAJ,C;;MAEb,OAAO,I;IACX,C;kGAEA,yB;MAAA,qD;MAAA,gE;MAAA,gD;MAAA,uC;QA
MI,IAAI,mBAAJ,C;UACI,OAAO,W;QACX,eAAe,+BAaA,cAAb,C;QACf,OAAO,QAAS,cAAhB,C;UACI,IAAI,C
AAC,UAAU,QAAS,WAAAnB,CAAL,C;YACI,QAAS,O;YACT,mBAAmB,iBAAO,QAAS,YAAhB,I;YACnB,IAAI
,iBAAGB,CAApB,C;AAuB,OAAO,W;YACI,kBAA3B,eAAa,YAAb,C;YACH,OAAgB,kBAAhB,C;cACI,sBAaA,
eAAb,C;YAFR,OF7uBD,W;;;QEmvBP,OAAO,iB;MACX,C;KArBA,C;0FAuBA,yB;MAAA,+D;MAAA,uC;QAOi
B,Q;QADb,WAAW,gB;QACE,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,IAAI,CAAC,UAAU,IAAV,CAAL,C;Y
ACI,K;UACJ,IAAK,WAAI,IAAJ,C;;QAET,OAAO,I;MACX,C;KAbA,C;;MAwBI,IAAI,wCAAsB,kBAAQ,CAAI
C,C;QA AqC,OAAO,mB;MAC5C,WAAW,0B;MACN,WAAI,IAAK,C;MACL,OAAO,I;IACX,C;oFAEA,yB;MA
AA,oD;MJzwBA,sC;MCjFA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,C
AAAd,C;MAAA,C;MAD5C,kF;MDiFA,uBAOe,yB;QArEf,8D;;UAqEe,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAA
c,SA4DgB,CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CAA3B,C;UA4DI,C;QAAA,C;OAAA,C;MIkwBf,sC;QAMI

,IAAI,iBAAO,CAAX,C;UAAc,iDJxwBH,iBIwwBsB,QJxwBtB,CIwwBG,E;;MACIB,C;KAPA,C;wGASA,yB;MAAA,oD;MJ/vBA,sC;MCpGA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,C;CAAd,C;MAAA,C;MAD5C,kF;MDoGA,iCAOe,yB;QAxFf,8D;;UAwFe,uB;YAAU,eAAsB,gB;YAAAtB,OA/Ed,cAAc,SA+EgB,CA/EhB,CAAd,EAA2B,SA+EM,CA/EN,CAA3B,C;UA+EI,C;QAAA,C;OAAA,C;MIwwBf,sC;QAMI,IAAI,iBAAO,CAAX,C;UAAc,iDJ9vBH,2BI8vBgC,QJ9vBhC,CI8vBG,E;;MACIB,C;KAPA,C;;MAeI,sBAAS,cAAT,C;IACJ,C;;MAWgB,Q;MAHZ,IAAI,oCAAJ,C;QACI,IAAI,kBAAQ,CAAZ,C;UAAe,OAAy,SAAL,SAAK,C;QAEwB,kBAA3C,sBC53BsD,sBD43BtD,uB;QAAmD,mB;QAA3D,OAAoE,OFxzBjE,WEwzBiE,C;;MAEjD,kBAAhB,0B;MAAwB,oB;MAA/B,OF1zBO,W;IE2zBX,C;wFAEA,yB;MAAA,wD;MJlzBA,sC;MCjFA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,C;CAAd,C;MAAA,C;MAD5C,kF;MDiFA,uBAOe,yB;QArEf,8D;;UAqEe,uB;YAAU,eAAsB,gB;YAAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CAA3B,C;UA4DI,C;QAAA,C;OAAA,C;MI2yBf,sC;QAMI,OAAO,mDjzBI,iBlizBiB,QJzBjB,ClizBJ,E;MACX,C;KAPA,C;4GASA,yB;MAAA,wD;MJxyBA,sC;MCpGA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,C;CAAd,C;MAAA,C;MAD5C,kF;MDoGA,iCAOe,yB;QAxFf,8D;;UAwFe,uB;YAAU,eAAsB,gB;YAAAtB,OA/Ed,cAAc,SA+EgB,CA/EhB,CAAd,EAA2B,SA+EM,CA/EN,CAA3B,C;UA+EI,C;QAAA,C;OAAA,C;MIiyBf,sC;QAMI,OAAO,mDjvyBI,2B1uyB2B,QJvyB3B,CluyBJ,E;MACX,C;KAPA,C;;MAeI,OAAO,wBAAW,cAAAX,C;IACX,C;;MAWe,Q;MAHX,IAAI,oCAAJ,C;QACG,IAAI,kBAAQ,CAAZ,C;UAAe,OAAy,SAAL,SAAK,C;QAEe,kBAAIC,sBCr6BuD,sBDq6BvD,uB;QAA0C,iC;QAAID,OAAyE,OFj2BrE,WEi2BqE,C;;MAErD,kBAAhB,0B;MAAwB,mC;MAA/B,OFn2BO,W;IEo2BX,C;;MAQoB,UACL,M;MAHX,aAAa,oBAAa,cAAb,C;MACb,YAAY,C;MACI,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,OAAO,cAAP,EAAO,sBAAP,YAAkB,O;;MACTB,OAAO,M;IACX,C;;MAQoB,UACL,M;MAHX,aAAa,cAAU,cAAV,C;MACb,YAAY,C;MACI,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,OAAO,cAAP,EAAO,sBAAP,YAAkB,O;;MACTB,OAAO,M;IACX,C;;MAQoB,UACL,M;MAHX,aAAa,iBAAU,cAAV,C;MACb,YAAY,C;MACI,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,oC;QACZ,OAAO,cAAP,EAAO,sBAAP,YAAkB,O;;MACTB,OAAO,M;IACX,C;;MAQoB,UACL,M;MAHX,aAAa,iBAAU,cAAV,C;MACb,YAAY,C;MACI,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,OAAO,cAAP,EAAO,sBAAP,YAAkB,O;;MACTB,OAAO,M;IACX,C;;MAQoB,UACL,M;MAHX,aAAa,eAAS,cAAT,C;MACb,YAAY,C;MACI,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,OAAO,cAAP,EAAO,sBAAP,YAAkB,O;;MACTB,OAAO,M;IACX,C;;MAQoB,UACL,M;MAHX,aAAa,eAAW,cAAAX,C;MACb,YAAY,C;MACI,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,OAAO,cAAP,EAAO,sBAAP,YAAkB,O;;MACTB,OAAO,M;IACX,C;0FAEA,yB;MAAA,kF;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,uC;QASI,eAAwD,cAAzC,YAAY,mCAAwB,EAAXB,CAAZ,CAAYC,EAAC,EAAD,C;QACjD,kBAAY,mBAAoB,QAAPB,C;QA+DH,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,WAhE8C,SAGe/B,CAAU,OAAV,C;UZhfnB,wBAAI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;QYgbA,OAkEO,W;MAjEX,C;KAXA,C;+FAAA,yB;MAAA,kF;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAQI,eAAwD,cAAzC,YAAY,mCAAwB,EAAXB,CAAZ,CAAYC,EAAC,EAAD,C;QACjD,kBAAC,mBAAoB,QAAPB,C;QAuBL,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,WAAy,aAxBoC,WAwBhC,CAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;QAxBhB,OA0BO,W;MAzBX,C;KAXA,C;+FAAA,yB;MAAA,kF;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,yD;QAQI,eAAwD,cAAzC,YAAY,mCAAwB,EAAXB,CAAZ,CAAYC,EAAC,EAAD,C;QACjD,kBAAC,mBAAoB,QAAPB,C;QAyBL,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,WAAy,aA1BoC,WA0BhC,CAAY,OAAZ,CAAJ,EA1BiD,cA0BvB,CAAe,OAAf,CAA1B,C;;QA1BhB,OA4BO,W;MA3BX,C;KAVA,C;;MAoBoB,Q;MAAA,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,WAAy,aAAI,YAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;MAEhB,OAAO,W;IACX,C;;MAUoB,Q;MAAA,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,WAAy,aAAI,YAAY,OAAZ,CAAJ,EAA0B,eAAe,OAAf,CAA1B,C;;MAEhB,OAAO,W;IACX,C;;MASoB,Q;MAAA,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,WAAe,UAAU,OAAV,C;QZhfnB,wBAAI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;MYkFA,OAAO,W;IACX,C;gGAEA,yB;MAAA,kF;MAAA,0D;MAAA,yD;MAAA,uE;MAAA,2C;QAYI,aAAa,mBAA6D,cAAzC,YAAY,mCAAwB,EAAXB,CAAZ,CAAYC,EAAC,EAAD,CAA7D,C;QAYG,Q;QAAA,2B;QAAhB,OAAGB

,cAAhB,C;UAAgB,yB;UAXO,MAYP,aAAI,OAAJ,EAZe,aAYF,CAAc,OAAd,CAAb,C;;QAZhB,OAAuB,M;MAC
3B,C;KAdA,C;;MAwBoB,Q;MAAA,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACZ,WAAY,aAAI,OAAJ,EA
Aa,cAAc,OAAd,CAAb,C;;MAEhB,OAAO,W;IACX,C;;MAMiB,Q;MAAA,2B;MAAb,OAAa,cAAb,C;QAAa,sB;
QACT,WAAY,WAAl,IAAJ,C;;MAEhB,OAAO,W;IACX,C;;MAMI,OAAO,0BAAa,eAAW,YAAY,mCAAwB,EA
AxB,CAAZ,CAAX,CAAb,C;IACX,C;;MAOqB,IAAN,I;MADX,IAAI,oCAAJ,C;QACW,QAAM,cAAN,C;eACH,C
;YAAK,kB;YAAL,K;eACA,C;YAAK,cAAW,8BAAJ,GAakB,sBAAl,CAAJ,CAAlB,GAA8B,oBAAW,OAAhD,C
;YAAL,K;kBACa,uBAAL,SAAK,C;YAHV,K;;QAAP,W;;MAMJ,OAA4B,qBAAhB,gBAAL,SAAK,CAAGB,C;IA
ChC,C;;MAMI,IAAI,oCAAJ,C;QACI,OAAy,gBAAL,SAAK,C;MACHB,OAAO,0BAAa,gBAAb,C;IACX,C;;MA
MI,OAAO,iBAAU,SAAV,C;IACX,C;;MASqB,IAAN,I;MADX,IAAI,oCAAJ,C;QACW,QAAM,cAAN,C;eACH,C;
YAAK,iB;YAAL,K;eACA,C;YAAK,aAAU,8BAAJ,GAakB,sBAAK,CAAL,CAAlB,GAA+B,oBAAW,OAAhD,C;
YAAL,K;kBACQ,iCAAA,qBAAiB,YAAY,cAAZ,CAAJB,CAAb,C;YAhL,K;;QAAP,W;;MAMJ,OAAwC,oBAAJc
,0BAAa,sBAAb,CAAIc,C;IAC5C,C;sFAEA,yB;MAAA,+D;MAOA,gD;MAPA,uC;QAIW,kBAAU,gB;QAOD,Q;
QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,WAR6B,SAQIB,CAAU,OAAV,C;UACC,OAAZ,WAA
Y,EAAO,IAAP,C;;QAThB,OAWO,W;MAVX,C;KALA,C;0FAOA,yB;MAAA,gD;MAAA,oD;QAIoB,Q;QAAA,2
B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,WAAW,UAAU,OAAV,C;UACC,OAAZ,WAAy,EAAO,IAAP,
C;;QAEhB,OAAO,W;MACX,C;KATA,C;uFAWA,yB;MAAA,wE;MAyBA,+D;MAzBA,yC;QASW,kBAAU,oB;Q
AyBD,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,UA1BiD,WA0BvC,CAAY,OAAZ,C;UZj8BP,
U;UADP,YYm8Be,WZn8BH,WYm8BwB,GZn8BxB,C;UACL,IAAI,aAAJ,C;YACH,aYi8BuC,gB;YAA5B,WZh8
BX,aYg8BgC,GZh8BhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UY67BA,iB;UACA,IAAK,WAAl,OAAJ,C;;QA5B
T,OA8BO,W;MA7BX,C;KAVA,C;uFAYA,yB;MAAA,wE;MA8BA,+D;MA9BA,yD;QAUW,kBAAU,oB;QA8BD
,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,UA/BiD,WA+BvC,CAAY,OAAZ,C;UZn9BP,U;UA
DP,YYq9Be,WZr9BH,WYq9BwB,GZr9BxB,C;UACL,IAAI,aAAJ,C;YACH,aYm9BuC,gB;YAA5B,WZl9BX,aYk
9BgC,GZl9BhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UY+8BA,iB;UACA,IAAK,WajCyD,cAiCrD,CAAe,OAaf,
CAAJ,C;;QAJCT,OAmCO,W;MAICX,C;KAXA,C;0FAaA,yB;MAAA,+D;MAAA,sD;QASoB,Q;QAAA,2B;QAAh
B,OAAGB,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAAY,OAAZ,C;UZj8BP,U;UADP,YYm8Be,WZn8BH,WYm8B
wB,GZn8BxB,C;UACL,IAAI,aAAJ,C;YACH,aYi8BuC,gB;YAA5B,WZh8BX,aYg8BgC,GZh8BhC,EAAS,MAAT
,C;YACA,e;;;YAEA,c;;UY67BA,iB;UACA,IAAK,WAAl,OAAJ,C;;QAET,OAAO,W;MACX,C;KafA,C;2FAiBA,
yB;MAAA,+D;MAAA,sE;QAUoB,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAAY,O
AAZ,C;UZn9BP,U;UADP,YYq9Be,WZr9BH,WYq9BwB,GZr9BxB,C;UACL,IAAI,aAAJ,C;YACH,aYm9BuC,gB
;YAA5B,WZl9BX,aYk9BgC,GZl9BhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UY+8BA,iB;UACA,IAAK,WAAl,e
AAe,OAaf,CAAJ,C;;QAET,OAAO,W;MACX,C;KahBA,C;4FAkBA,yB;MAAA,kC;MAAA,4C;;QAQW,sC;QA
AA,8C;MARX,C;MAAA,gE;QASqD,OAAA,oBAAGB,W;MAAhB,C;MATrD,oE;QAU4C,gCAAY,OAAZ,C;MA
AA,C;MAV5C,gF;MAAA,yC;QAQI,2D;MAIJ,C;KAZA,C;8EAca,yB;MAAA,kF;MAAA,gE;MAAA,uC;QAOW,
kBAAM,eAAa,mCAAwB,EAAXB,CAAb,C;QAqEA,Q;QAAA,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAA
Y,WAtEwC,SAsEpC,CAAU,IAAV,CAAJ,C;;QAtEhB,OAuEO,W;MatEX,C;KARA,C;4FAUA,yB;MAAA,kF;MA
AA,gE;MA+BA,wE;MA/BA,uC;QAOW,kBAAa,eAAa,mCAAwB,EAAXB,CAAb,C;QAqCP,gB;QADb,YAAY,C;
QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WajC+C,SAiC3C,CAAU,oBAAmB,cAAnB,EAAMB,
sBAAnB,UAAV,EAAuC,IAAvC,CAAJ,C;;QAJChB,OakCO,W;MAjCX,C;KARA,C;0GAUA,yB;MAAA,+D;MA
wQA,wE;MAxQA,uC;QAOW,kBAAoB,gB;QAwQd,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa
,sB;UA9PsB,U;UAAA,cAVQ,SAUR,CA8PT,oBAAmB,cAAnB,EAAMB,sBAAnB,UA9PS,EA8PoB,IA9PpB,W;Y
AA6C,6B;;;QAVhF,OAWO,W;MAVX,C;KARA,C;8GAUA,yB;MA8PA,wE;MA9PA,oD;QAqQiB,gB;QADb,YA
AY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UA9PsB,U;UAAA,wBA8PT,oBAAmB,cAAnB,EAAMB,sBAA
nB,UA9PS,EA8PoB,IA9PpB,W;YAA6C,6B;;;QACHf,OAAO,W;MACX,C;KATA,C;+FAWA,yB;MAAA,wE;MA
AA,oD;QAQiB,UACoC,M;QAFjD,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAAl,U
AAU,oBAAmB,cAAnB,EAAMB,sBAAnB,UAAV,EAAuC,IAAvC,CAAJ,C;;QACHB,OAAO,W;MACX,C;KAXA,
C;4FAaA,yB;MAAA,+D;MAAA,uC;QAKW,kBAAa,gB;QA8NJ,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAg
B,yB;UAtNK,U;UAAA,cARe,SAQf,CAsNQ,OAtNR,W;YAAsC,6B;;;QAR3D,OASO,W;MARX,C;KANA,C;gGA
QA,yB;MAAA,oD;QA2NoB,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UAtNK,U;UAAA,wBAsNQ,O

AtNR,W;YAAc,6B;;;QAC3D,OAAO,W;MACX,C;KAPA,C;;MAciB,Q;MAAA,2B;MAAb,OAAa,cAAb,C;QAA
a,sB;QACT,WAAy,WAAI,UAAU,IAAV,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAO4B,mB;QAAE,gC;MAAF,
C;IAAA,C;;MAAxB,OAAO,qBAAiB,6BAAjB,C;IACX,C;;MAQI,OAA2B,SAAf,eAAL,SAAK,CAAe,C;IAC/B,C;
4FAEA,yB;MAAA,2D;MAAA,+D;MAAA,sC;QASc,Q;QAFV,UAAU,c;QACV,WAAW,gB;QACD,2B;QAAV,O
AAU,cAAV,C;UAAU,mB;UACN,UAAU,SAAS,CAAT,C;UACV,IAAI,GAAI,WAAI,GAJ,CAAR,C;YACI,IAA
K,WAAI,CAAJ,C;;QAEb,OAAO,I;MACX,C;KAFa,C;;MAuBI,UAAe,eAAL,SAAK,C;MACX,YAAJ,GAAI,EAA
U,KAAV,C;MACJ,OAAO,G;IACX,C;;MAQI,UAAe,eAAL,SAAK,C;MACX,YAAJ,GAAI,EAAU,KAAV,C;MAC
J,OAAO,G;IACX,C;;MAQiB,IAAN,I;MACH,kBADs,SACT,c;QAAoB,4BAAc,SAAd,C;;QACZ,iCAAa,sBAAb,C
;MAFZ,W;IAIJ,C;;MAUI,UAAe,eAAL,SAAK,C;MACX,OAAJ,GAAI,EAAO,KAAP,C;MACJ,OAAO,G;IACX,C;
8EAEA,yB;MAAA,gD;MAAA,uC;QAOoB,Q;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,OAAO,I;QAC5B,2B;
QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CAAC,UAAU,OAAV,CAAL,C;YAAyB,OAAO,K;;QACtD,
OAAO,I;MACX,C;KATA,C;;MAiBI,IAAI,oCAAJ,C;QAAwB,OAAO,CAAC,mB;MACHC,OAAO,oBAAW,U;IA
CtB,C;+EAEA,yB;MAAA,gD;MAAA,uC;QAOoB,Q;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,OAAO,K;QA
C5B,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,I;;QACrD
,OAAO,K;MACX,C;KATA,C;;MAiBoB,Q;MAFhB,IAAI,oCAAJ,C;QAAwB,OAAO,c;MAC/B,YAAy,C;MACI,2
B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,oBAAmB,qBAAnB,EAAMB,KAAAnB,E;;MACtB,OAAO,K;IA
CX,C;;MAOI,OAAO,c;IACX,C;mFAEA,yB;MAAA,gD;MAAA,wE;MAAA,uC;QAMoB,Q;QAFhB,IAAI,wCAAs
B,mBAA1B,C;UAAqC,OAAO,C;QAC5C,YAAy,C;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IA
AI,UAAU,OAAV,CAAJ,C;YAAwB,oBAAmB,qBAAnB,EAAMB,KAAAnB,E;;QAC9C,OAAO,K;MACX,C;KARA
,C;;MAeoB,Q;MADhB,kBAAkB,O;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,UAAU,WA
AV,EAAuB,OAAvB,C;;MACpC,OAAO,W;IACX,C;8FAEA,yB;MAAA,wE;MAAA,gD;QASoB,UAAiD,M;QAFj
E,YAAy,C;QACZ,kBAAkB,O;QACF,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,cAAc,UAAU,oBAAM
B,cAAAnB,EAAMB,sBAAnB,UAAV,EAAuC,WAAvC,EAAoD,OAApD,C;;QACpC,OAAO,W;MACX,C;KAXA,C
;;MAiBI,kBAAkB,O;MACIB,IAAI,CAAC,mBAAL,C;QACI,eAAe,+BAAa,cAAb,C;QACf,OAAO,QAAS,cAAhB,
C;UACI,cAAc,UAAU,QAAS,WAAAnB,EAA+B,WAA/B,C;;MAGtB,OAAO,W;IACX,C;;MASI,kBAAkB,O;MAC
IB,IAAI,CAAC,mBAAL,C;QACI,eAAe,+BAAa,cAAb,C;QACf,OAAO,QAAS,cAAhB,C;UACI,YAAy,QAAS,gB
;UACrB,cAAc,UAAU,KAAV,EAAiB,QAAS,WAA1B,EAAc,WAAtC,C;;MAGtB,OAAO,W;IACX,C;;MAOoB,
Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,OAAO,OAAP,C;;IAC1B,C;oGAEA,yB;MAAA,wE
;MAAA,oC;QAOiB,UAAgC,M;QAD7C,YAAy,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UAAM,OAAO,oB
AAmB,cAAAnB,EAAMB,sBAAnB,UAAP,EAAoC,IAApC,C;;MACvB,C;KARA,C;;MAiBI,eAAe,oB;MACf,IAAI,
CAAC,QAAS,UAAc,C;QAAyB,OAAO,I;MACHC,UAAU,QAAS,O;MACnB,IAAQ,QAAJ,GAAI,CAAR,C;QAAi
B,OAAO,G;MACxB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAM,QAAF,CAAE,CAAN,C;U
AAe,OAAO,C;QACtB,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MASI,eAAe,oB;M
ACf,IAAI,CAAC,QAAS,UAAc,C;QAAyB,OAAO,I;MACHC,UAAU,QAAS,O;MACnB,IAAQ,QAAJ,GAAI,CAA
R,C;QAAiB,OAAO,G;MACxB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAM,QAAF,CAAE,C
AAN,C;UAAe,OAAO,C;QACtB,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAMI,e
AAe,oB;MACf,IAAI,CAAC,QAAS,UAAc,C;QAAyB,OAAO,I;MACHC,UAAU,QAAS,O;MACnB,OAAO,QAAS,
UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAI,sBAAM,CAAN,KAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;I
ACX,C;;MAQI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAc,C;QAAyB,OAAO,I;MACHC,cAAc,QAAS,O;MACvB
,IAAI,CAAC,QAAS,UAAc,C;QAAyB,OAAO,O;MACHC,eAAe,SAAS,OAAT,C;;QAEX,QAAQ,QAAS,O;QACjB
,QAAQ,SAAS,CAAT,C;QACR,IAAI,2BAAW,CAAX,KAAJ,C;UACI,UAAU,C;UACV,WAAW,C;;MAED,QAA
T,QAAS,W;MACIB,OAAO,O;IACX,C;;MAMI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAc,C;QAAyB,OAAO,I;
MACHC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAI,UAAW,SAAQ
,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MASI,eAAe,oB;
MACf,IAAI,CAAC,QAAS,UAAc,C;QAAyB,OAAO,I;MACHC,UAAU,QAAS,O;MACnB,IAAQ,QAAJ,GAAI,CA
AR,C;QAAiB,OAAO,G;MACxB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAM,QAAF,CAAE,
CAAN,C;UAAe,OAAO,C;QACtB,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MASI,e
AAe,oB;MACf,IAAI,CAAC,QAAS,UAAc,C;QAAyB,OAAO,I;MACHC,UAAU,QAAS,O;MACnB,IAAQ,QAAJ,G

AAI,CAAR,C;QAAiB,OAAO,G;MACxB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAM,QAAF,CAAE,CAAN,C;UAAe,OAAO,C;QACtB,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAMI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAo,C;QAAyB,OAAO,I;MACHC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAI,sBAAM,CAAN,KAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAo,C;QAAyB,OAAO,I;MACHC,cAAc,QAAS,O;MACvB,IAAI,CAAC,QAAS,UAAo,C;QAAyB,OAAO,O;MACHC,eAAe,SAAS,OAAT,C;;QAEX,QAAQ,QAAS,O;QACjB,QAAQ,SAAS,CAAT,C;QACR,IAAI,2BAAW,CAAX,KAAJ,C;UACI,UAAU,C;UACV,WAAW,C;;MAED,QAAT,QAAS,W;MACIB,OAAO,O;IACX,C;;MAMI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAo,C;QAAyB,OAAO,I;MACHC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQI,IAAI,oCAAJ,C;QAAwB,OAAO,mB;MAC/B,OAAO,CAAC,oBAAW,U;IACvB,C;iFAEA,yB;MAAA,gD;MAAA,uC;QAOoB,Q;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,OAAO,I;QAC5B,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,UAAU,OAAs,CAAJ,C;YAAwB,OAAO,K;;QACrD,OAAO,I;MACX,C;KATA,C;;MAGbM,C,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,OAAO,OAAP,C;;MAArC,gB;IACJ,C;oFAEA,yB;MAAA,4F;MAAA,uC;QAII,eAAe,SAAK,W;QACpB,IAAI,CAAC,QAAS,UAAo,C;UAAyB,MAAM,mCAA8B,oCAA9B,C;QAC/B,kBAAqB,QAAS,O;QAC9B,OAAO,QAAS,UAAhB,C;UACI,cAAc,UAAU,WAAV,EAAuB,QAAS,OAAhC,C;;QAEIB,OAAO,W;MACX,C;KAXA,C;kGAaA,yB;MAAA,4F;MAAA,wE;MAAA,uC;QAYmD,Q;QAL/C,eAAe,SAAK,W;QACpB,IAAI,CAAC,QAAS,UAAo,C;UAAyB,MAAM,mCAA8B,oCAA9B,C;QAC/B,YAAy,C;QACZ,kBAAqB,QAAS,O;QAC9B,OAAO,QAAS,UAAhB,C;UACI,cAAc,UAAU,oBAAmB,YAAnB,EAAmB,oBAAmB,QAAs,EAAuB,WAAvC,EAAoD,QAAS,OAA7D,C;;QAEIB,OAAO,W;MACX,C;KAFa,C;8FAiB A,yB;MAAA,4F;MAAA,uC;QAII,eAAe,+BAAa,cAAb,C;QACf,IAAI,CAAC,QAAS,cAAc,C;UACI,MAAM,mCAA8B,8BAA9B,C;QACV,kBAAqB,QAAS,W;QAC9B,OAAO,QAAS,cAAhB,C;UACI,cAAc,UAAU,QAAS,WAAmB,EAA+B,WAA/B,C;;QAEIB,OAAO,W;MACX,C;KAZA,C;4GAcA,yB;MAAA,4F;MAAA,uC;QAOI,eAAe,+BAaA,cAAb,C;QACf,IAAI,CAAC,QAAS,cAAc,C;UACI,MAAM,mCAA8B,8BAA9B,C;QACV,kBAAqB,QAAS,W;QAC9B,OAAO,QAAS,cAAhB,C;UACI,YAAy,QAAS,gB;UACrB,cAAc,UAAU,KAAV,EAAiB,QAAS,WAA1B,EAA5C,WAAtC,C;;QAEIB,OAAO,W;MACX,C;KAhBA,C;;MAuBoB,Q;MADhB,UAAe,C;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QACZ,YAAO,SAAS,OAAT,CAAP,I;;MAEJ,OAAO,G;IACX,C;;MAOoB,Q;MADhB,UAAkB,G;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QACZ,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;IACX,C;;MAMoB,UAMT,M;MANS,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QACZ,IAAI,eAAJ,C;UACI,MAAM,gCAAyB,2BAAwB,SAAXB,MAAZB,C;;MAId,OAAO,mE;IACX,C;;MAMoB,UAMT,M;MANS,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QACZ,IAAI,eAAJ,C;UACI,MAAM,gCAAyB,2BAAwB,SAAXB,MAAZB,C;;MAId,OAAO,+D;IACX,C;;MAaI,OAAO,oBAAS,IAAT,EAAe,IAAf,EAA5C,IAAtC,C;IACX,C;;MAkBI,OAAO,sBAAS,IAAT,EAAe,IAAf,EAA5C,IAAtC,EAAwD,SAAXD,C;IACX,C;;MAMI,aAAa,iBAAa,mCAAwB,EAAxB,CAAb,C;MACb,kBAAc,KAAc,C;MAvrCgB,Q;MAAA,OAwrCT,SAxrCS,W;MAAhB,OAAgB,cAAhB,C;QAAGB,2B;QAAU,oB;QAWrCK,IAAI,CAAC,SAAD,IAAY,OAxrCX,SAwrCW,UAAhB,C;UAAiC,YAAU,I;UAA3C,mBAAiD,K;;UAAjD,mBAA8D,I;;QAXrCvE,qB;UAWrCD,MAxrCqC,WAAI,SAAJ,C;;MAwrC1D,OAAqB,M;IACzB,C;;MASI,IAAI,QnBg8IG,YAAQ,CmBh8If,C;QAAwB,OAAy,SAAL,SAAK,C;MACpC,YAAqB,UAAT,QAAS,C;MAjuCd,kBAAy,gB;MASBH,Q;MAAA,OA4sCT,SA5sCS,W;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,CA4sCF,qBA5sCa,OA4sCb,CA5sCF,C;UAAyB,WAAy,WAAI,OAAJ,C;;MA4sC3D,OA3sCO,W;IA4sCX,C;;MASI,YAAqB,gCAAT,QAAS,EAAgC,SAAhC,C;MACrB,IAAI,KAAM,UAAV,C;QACI,OAAy,SAAL,SAAK,C;MA9uCT,kBAAy,gB;MASBH,Q;MAAA,OAytCT,SAztCS,W;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,CAytCF,qBAztCa,OAytCb,CAztCF,C;UAAyB,WAAy,WAAI,OAAJ,C;;MAytC3D,OAxtCO,W;IAytCX,C;;MASI,YAAqB,YAAT,QAAS,C;MACrB,IAAI,KAAM,UAAV,C;QACI,OAAy,SAAL,SAAK,C;MA3vCT,kBAAy,gB;MASBH,Q;MAAA,OA5uCT,SAtuCS,W;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,CA5uCF,qBAtuCa,OA5uCb,CAtuCF,C;UAAyB,WAAy,WAAI,OAAJ,C;;MA5uC3D,OAruCO,W;IA5uCX,C;8FAEA,yB;MAAA,8C;MAAA,qC;QAKI,OAAO,iBAAM,OAAN,C;MACX,C;KANA,C;0FAQA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAQoB,Q;QAFhB,YAAy,gB;QACZ,aAAa,gB;QACG,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,OAAV,CAAJ,C;YACI,KAAM,WAAI,OAAJ,C;;YAEN,MAAO,WAAI,OAAJ,C;;QAGf,OAAO,cAAK,KAAL,

EAAY,MAAZ,C;MACX,C;KAhBA,C;;MAsBI,IAAI,oCAAJ,C;QAAwB,OAAy,OAAL,SAAK,EAAK,OAAL,C;
MACpC,aAAa,gB;MACN,OAAP,MAAO,EAAO,SAAP,C;MACP,MAAO,WAAI,OAAJ,C;MACP,OAAO,M;IAC
X,C;;MAMI,aAAa,iBAAa,iBAAO,CAAP,IAAb,C;MACb,MAAO,gBAAO,SAAP,C;MACP,MAAO,WAAI,OAAJ,
C;MACP,OAAO,M;IACX,C;;MAMI,IAAI,oCAAJ,C;QAAwB,OAAy,OAAL,SAAK,EAAK,QAAL,C;MACpC,aA
Aa,gB;MACN,OAAP,MAAO,EAAO,SAAP,C;MACA,SAAP,MAAO,EAAO,QAAP,C;MACP,OAAO,M;IACX,C;;
MAMI,aAAa,iBAAa,SAAK,KAAL,GAAY,QAAS,OAARb,IAAb,C;MACb,MAAO,gBAAO,SAAP,C;MACA,SA
AP,MAAO,EAAO,QAAP,C;MACP,OAAO,M;IACX,C;;MAMI,IAAI,oCAAJ,C;QAAwB,OAAy,OAAL,SAAK,EA
AK,QAAL,C;MACpC,aAAa,gB;MACN,OAAP,MAAO,EAAO,SAAP,C;MACA,OAAP,MAAO,EAAO,QAAP,C;
MACP,OAAO,M;IACX,C;;MAMI,IAAI,mCAAJ,C;QACI,aAAa,iBAAa,SAAK,KAAL,GAAY,QAAS,KAARb,IA
Ab,C;QACb,MAAO,gBAAO,SAAP,C;QACP,MAAO,gBAAO,QAAP,C;QACP,OAAO,M;;;QAEp,eAAa,iBAAa,S
AAb,C;QACN,OAAP,QAAO,EAAO,QAAP,C;QACP,OAAO,Q;;IAEf,C;;MAMI,aAAa,gB;MACN,OAAP,MAAO,
EAAO,SAAP,C;MACA,SAAP,MAAO,EAAO,QAAP,C;MACP,OAAO,M;IACX,C;;MAMI,aAAa,iBAAa,SAAK,
KAAL,GAAY,EAaz,IAAb,C;MACb,MAAO,gBAAO,SAAP,C;MACA,SAAP,MAAO,EAAO,QAAP,C;MACP,O
AAO,M;IACX,C;4FAEA,yB;MAAA,4C;MAAA,qC;QAKI,OAAO,gBAAK,OAAL,C;MACX,C;KANA,C;8FAQA,
yB;MAAA,4C;MAAA,qC;QAKI,OAAO,gBAAK,OAAL,C;MACX,C;KANA,C;;MAwB+C,oB;QAAA,OAAy,C;
MAAG,8B;QAAA,iBAA0B,K;MACpF,oBAAoB,IAApB,EAA0B,IAA1B,C;MACA,IAAI,0CAAwB,8BAA5B,C;Q
ACI,eAAe,SAAK,K;QACpB,aAAa,iBAAmB,CAAC,WAAW,IAAX,GAaKb,CAAIb,IAAD,IAAwB,IAAxB,IAAn
B,C;QACb,gBAAy,CAAz,C;QACA,OAAO,UAAQ,QAaf,C;UACI,iBAAsB,eAAL,IAAK,EAAa,WAAW,OAAx,
IAAb,C;UACtB,IAAI,aAAa,IAAb,IAAqB,CAAC,cAA1B,C;YAA0C,K;Ub/8DID,WAAW,iBag9Da,Ubh9Db,C;U
WDX,mBAAc,CAAd,YEi9DwB,UFj9DxB,Y;YXE6B,ea+8DS,sBFh9D3B,OEg9DgC,GAAK,OAAL,IAAL,Cb/8D
T,C;;Ua+8DrB,MAAO,Wb98DR,Ia88DQ,C;UACP,oBAAS,IAAT,I;;QAEJ,OAAO,M;;MAEX,eAAa,gB;MACiE,k
BAA9E,iBAAiB,oBAAjB,EAA6B,IAA7B,EAAMc,IAAnC,EAAYc,cAAzC,EAAuE,KAAvE,C;ME7kEA,OAAgB,
qBAAhB,C;QAAgB,gC;QF8kEL,mBE9kEqB,OF8kErB,C;;MAEX,OAAO,Q;IACX,C;;MAoBkD,oB;QAAA,OAA
Y,C;MAAG,8B;QAAA,iBAA0B,K;MACvF,oBAAoB,IAApB,EAA0B,IAA1B,C;MACA,IAAI,0CAAwB,8BAA5B
,C;QACI,eAAe,SAAK,K;QACpB,aAAa,iBAAa,CAAC,WAAW,IAAX,GAaKb,CAAIb,IAAD,IAAwB,IAAxB,IA
Ab,C;QACb,eAAa,kBAAc,SAAd,C;QACb,YAAy,C;QACZ,OAAO,QAAQ,QAaf,C;UACI,QAAO,cAAK,KAAL,
EAA2B,eAAd,QAAQ,IAAR,IAAc,EAAa,QAAb,CAA3B,C;UACP,IAAI,CAAC,cAAD,IAAmB,QAAO,KAAP,GA
Ac,IAArC,C;YAA2C,K;UAC3C,MAAO,WAAI,UAAU,QAAV,CAAJ,C;UACP,gBAAS,IAAT,I;;QAEJ,OAAO,M;
;MAEX,eAAa,gB;MACgE,kBAA7E,iBAAiB,oBAAjB,EAA6B,IAA7B,EAAMc,IAAnC,EAAYc,cAAzC,EAAuE,I
AAvE,C;MErnEA,OAAgB,qBAAhB,C;QAAgB,gC;QFsnEL,mBAAI,UEtnEiB,OFsnEjB,CAAJ,C;;MAEX,OAAO,
Q;IACX,C;;MAuBoB,gB;MAHhB,gBAXW,KAWW,O;MACtB,WAAW,iBDp/DJ,MAAK,KCo/DkB,mCAAwB,E
AAxB,CDp/DIB,EC0/D+C,SDp/D/C,CC0/DD,C;MACX,QAAQ,C;MACQ,2B;MAAhB,OAAgB,cAAhB,C;QAAg
B,yB;QACZ,IAAI,KAaK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhBqB,GAgBP,OAhBO,EAAAnB,KAgBqB,CAA
M,UAAAn,EAAM,kBAAN,SAhBF,CAGrB,C;;MAhBT,OAKBo,I;IAjBX,C;+EAEA,yB;MAAA,kF;MAAA,gE;M
DI/DA,iB;MCk/DA,8C;QAWoB,UAEsB,M;QALiC,gBAAgB,KAAM,O;QACtB,WAAW,eDp/DJ,MAAK,KCo/Dk
B,mCAAwB,EAAxB,CDp/DIB,EC0/D+C,SDp/D/C,CC0/DD,C;QACX,QAAQ,C;QACQ,2B;QAAhB,OAAgB,cAA
hB,C;UAAgB,yB;UACZ,IAAI,KAaK,SAAT,C;YAAoB,K;UACpB,IAAK,WAAI,UAAU,OAAV,EAAmB,MAAM
,UAAAn,EAAM,kBAAN,SAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;;MAoCI,YAAy,oB;MACZ,aAZ
W,KAYQ,W;MACnB,WAAW,iBDjhEJ,MAAK,KCihEkB,mCAAwB,EAAxB,CDjhEIB,ECihEqD,wBAbtD,Kaas
D,EAAwB,EAAxB,CDjhErD,CCihED,C;MACX,OAAO,KAAM,UAAAn,IAAmB,MAAO,UAAjC,C;QACI,IAAK,
WafqB,GAeP,KAAM,OAfC,EAeO,MAAO,OAfd,CAerB,C;;MAfT,OAIBo,I;IAhBX,C;+EAEA,yB;MAAA,kF;M
AAA,gE;MD9gEA,iB;MC8gEA,8C;QAQI,YAAy,oB;QACZ,aAAa,KAAM,W;QACnB,WAAW,eDjhEJ,MAAK,K
CihEkB,mCAAwB,EAAxB,CDjhEIB,ECihEqD,wBAAN,KAAM,EAAwB,EAAxB,CDjhErD,CCihED,C;QACX,O
AAO,KAAM,UAAAn,IAAmB,MAAO,UAAjC,C;UACI,IAAK,WAAI,UAAU,KAAM,OAAhB,EAAwB,MAAO,O
AA/B,CAAJ,C;;QAET,OAAO,I;MACX,C;KAfA,C;;MA0BW,sB;;QAaP,eAAe,oB;QACf,IAAI,CAAC,QAAS,UA
Ad,C;UAAyB,qBAAO,W;UAAP,uB;;QACzB,abhpEoD,gB;QaipEpD,cAAc,QAAS,O;QACvB,OAAO,QAAS,UA
AhB,C;UACI,WAAW,QAAS,O;UACpB,MAAO,WAnBkB,GAmBJ,OAnBI,EAmBK,IAnBL,CAMBiB,C;UACP,U
AAU,I;;QAEd,qBAAO,M;;;MATBP,yB;IACJ,C;8FAEA,yB;MAAA,qD;Mb1oEA,+D;Ma0oEA,uC;QAUI,eAAe,oB

;QACf,IAAI,CAAC,QAAS,UAAAd,C;UAAyB,OAAO,W;QACChC,abhpEoD,gB;QaipEpD,cAAc,QAAS,O;QACvB,
OAAO,QAAS,UAAhB,C;UACI,WAAW,QAAS,O;UACpB,MAAO,WAAI,UAAU,OAAV,EAAmB,IAAnB,CAAJ,
C;UACP,UAAU,I;QAEd,OAAO,M;MACX,C;KApBA,C;;MA8B6D,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SA
AuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,
YAAoC,I;MAGtN,Q;MAFhB,MAAO,gBAAO,MAAP,C;MACP,YAAY,C;MACI,2B;MAAhB,OAAgB,cAAhB,C;
QAAgB,yB;QACZ,IAAI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,CAAR,IAAa,S
AAS,KAA1B,C;UACW,gBAAP,MAAO,EAAc,OAAAd,EAAuB,SAAvB,C;;;UACJ,K;;MAEX,IAAI,SAAS,CAAT,I
AAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,MAAO,gBAAO,OAAp,C;MACP,OAAO,M;IA
CX,C;;MAUwC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAA
A,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAoC,I;MACjN,OAAO,oBAAO,sBAAP,EAAwB,S
AAxB,EAAmC,MAAnC,EAA2C,OAA3C,EAAoD,KAApD,EAA2D,SAA3D,EAA5E,SAAtE,CAAiF,W;IAC5F,C;;
MAOI,OAAO,S;IACX,C;;MAQoB,mB;QAAE,OAAA,eAAK,W;MAAP,C;IAAA,C;;MAAhB,oCAAgB,8BAAhB,
C;IACJ,C;;MASoB,Q;MAFhB,UAAkB,G;MACIB,YAAiB,C;MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;
QACZ,OAAO,O;QACP,oBAAmB,qBAAnB,EAAmB,KAAAnB,E;;MAEJ,OAAW,UAAS,CAAb,GAAgB,wCAAo,I
AAvB,GAAgC,MAAM,K;IACjD,C;;MASoB,Q;MAFhB,UAAkB,G;MACIB,YAAiB,C;MACD,2B;MAAhB,OAAg
B,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;QACP,oBAAmB,qBAAnB,EAAmB,KAAAnB,E;;MAEJ,OAAW,UAAS,
CAAb,GAAgB,wCAAo,IAAvB,GAAgC,MAAM,K;IACjD,C;;MASoB,Q;MAFhB,UAAkB,G;MACIB,YAAiB,C;
MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;QACP,oBAAmB,qBAAnB,EAAmB,KAAAnB
,E;;MAEJ,OAAW,UAAS,CAAb,GAAgB,wCAAo,IAAvB,GAAgC,MAAM,K;IACjD,C;;MASoB,Q;MAFhB,UAA
kB,G;MACIB,YAAiB,C;MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;QACP,oBAAmB,q
BAAnB,EAAmB,KAAAnB,E;;MAEJ,OAAW,UAAS,CAAb,GAAgB,wCAAo,IAAvB,GAAgC,MAAM,K;IACjD,C;
;MASoB,Q;MAFhB,UAAkB,G;MACIB,YAAiB,C;MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OA
AO,O;QACP,oBAAmB,qBAAnB,EAAmB,KAAAnB,E;;MAEJ,OAAW,UAAS,CAAb,GAAgB,wCAAo,IAAvB,GA
AgC,MAAM,K;IACjD,C;;MASoB,Q;MAFhB,UAAkB,G;MACIB,YAAiB,C;MACD,2B;MAAhB,OAAgB,cAAhB,
C;QAAgB,yB;QACZ,OAAO,O;QACP,oBAAmB,qBAAnB,EAAmB,KAAAnB,E;;MAEJ,OAAW,UAAS,CAAb,GA
AgB,wCAAo,IAAvB,GAAgC,MAAM,K;IACjD,C;;MAQoB,Q;MADhB,UAAe,C;MACC,2B;MAAhB,OAAgB,c
AAhB,C;QAAgB,yB;QACZ,YAAO,O;;MAEX,OAAO,G;IACX,C;;MAQoB,Q;MADhB,UAAe,C;MACC,2B;MAA
hB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,YAAO,O;;MAEX,OAAO,G;IACX,C;;MAQoB,Q;MADhB,UAAe,C;M
ACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,YAAO,OAAP,I;;MAEJ,OAAO,G;IACX,C;;MAQoB,Q;
MADhB,Y;MACgB,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,cAAO,OAAP,C;;MAEJ,OAAO,G;IACX,
C;;MAQoB,Q;MADhB,UAAiB,G;MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;;MAEX,O
AAO,G;IACX,C;;MAQoB,Q;MADhB,UAAkB,G;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAA
O,O;;MAEX,OAAO,G;IACX,C;;MGz0EI,OAAO,QAAM,CAAN,EAAS,QAAM,CAAN,EAAS,CAAT,EAAY,UA
AZ,CAAT,EAakC,UAAIC,C;IACX,C;;MAQI,OAAW,UAAW,SAAQ,CAAR,EAAW,CAAX,CAAX,IAA4B,CAA
hC,GAAmC,CAAnC,GAA0C,C;IACrD,C;;MAwGI,OAAO,QAAM,CAAN,EAAS,QAAM,CAAN,EAAS,CAAT,E
AAY,UAAZ,CAAT,EAakC,UAAIC,C;IACX,C;;MAQI,OAAW,UAAW,SAAQ,CAAR,EAAW,CAAX,CAAX,IA
A4B,CAAhC,GAAmC,CAAnC,GAA0C,C;IACrD,C;;MC5NI,IAAI,mBAAQ,CAAZ,C;QACI,OAAO,W;MACX,e
AAe,iBAAQ,W;MACvB,IAAI,CAAC,QAAS,UAAAd,C;QACI,OAAO,W;MACX,YAAY,QAAS,O;MACrB,IAAI,C
AAC,QAAS,UAAAd,C;QACI,OAAO,OhBwOiD,SgBxO1C,KhBwO+C,IAAL,EgBxO1C,KhBwOoD,MAAV,CgBx
OjD,C;;MACX,aAAa,iBAAsB,cAAtB,C;MACb,MAAO,WhBsOqD,SgBtOjD,KhBsOsD,IAAL,EgBtOjD,KhBsO2
D,MAAV,CgBtOrD,C;;QAEwB,kBAAhB,QAAS,O;QAApB,MAAO,WhBoOiD,SAAK,eAAL,EAAU,iBAAV,CgB
pOjD,C;;MACO,QAAT,QAAS,W;MACIB,OAAO,M;IACX,C;sFAEA,yB;MAAA,+D;MAOA,gD;MAPA,uC;QAI
W,kBAAU,gB;QAOD,Q;QAAA,OhB8QoE,iBAAQ,W;QgB9Q5F,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAR6B,
SAQIB,CAAU,OAAV,C;UACC,OAAZ,WAA,Y,EAAO,IAAP,C;;QAThB,OAwo,W;MAVX,C;KALA,C;0FAOA,
yB;MAAA,gD;MAAA,oD;QAIoB,Q;QAAA,OAAA,ShB8QoE,QAAQ,W;QgB9Q5F,OAAgB,cAAhB,C;UAAgB,y
B;UACZ,WAAW,UAAU,OAAV,C;UACC,OAAZ,WAA,Y,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KATA,C
;8EAWA,yB;MAAA,gE;MAAA,uC;QAOW,kBAAM,eAAa,cAAb,C;QAYBA,Q;QAAA,OhBuOuE,iBAAQ,W;Qg
BvO5F,OAAa,cAAb,C;UAAa,sB;UACT,WAA,Y,WA1BiB,SA0Bb,CAAU,IAAV,CAAJ,C;;QA1BhB,OA2BO,W;

MA1BX,C;KARA,C;4FAUA,yB;MAAA,+D;MAAA,uC;QAKW,kBAAa,gB;QA4EJ,Q;QAAA,OhB4KoE,iBAAQ,W;QgB5K5F,OAAgB,cAAhB,C;UAAgB,yB;UApEK,U;UAAA,cARe,SAQf,CAoEQ,OApER,W;YAAc,6B;;;QAR3D,OASO,W;MARX,C;KANA,C;gGAQA,yB;MAAA,oD;QAYeOB,Q;QAAA,OhB4KoE,iBAAQ,W;QgB5K5F,OAAgB,cAAhB,C;UAAgB,yB;UApEK,U;UAAA,wBAoEQ,OApER,W;YAAc,6B;;;QAC3D,OAAO,W;MACX,C;KAPA,C;;MAciB,Q;MAAA,OAAA,ShBuOuE,QAAQ,W;MgBvO5F,OAAa,cAAb,C;QAAa,sB;QACT,WAAy,WAAI,UAAU,IAAV,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MASoB,Q;MADhB,IAAI,mBAAJ,C;QAAe,OAAO,I;MACN,OAAA,ShB2NoE,QAAQ,W;MgB3N5F,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I;IACX,C;;MAQI,OAAO,CAAC,mB;IACZ,C;;MASoB,Q;MADhB,IAAI,mBAAJ,C;QAAe,OAAO,K;MACN,OAAA,ShBuMoE,QAAQ,W;MgBvM5F,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,I;;MACrD,OAAO,K;IACX,C;;MAOI,OAAO,c;IACX,C;;MAQoB,Q;MAFhB,IAAI,mBAAJ,C;QAAe,OAAO,C;MACtB,YAAy,C;MACI,OAAA,ShBqLoE,QAAQ,W;MgBrL5F,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,qB;;MAC9C,OAAO,K;IACX,C;;MAOoB,Q;MAAA,OAAA,ShB4KoE,QAAQ,W;MgB5K5F,OAAgB,cAAhB,C;QAAgB,yB;QAAM,OAAO,OAAp,C;;IAC1B,C;;MASmB,kBAAR,iB;MAAQ,gB;;QJ29Cf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAAd,C;UAAyB,eAAO,I;UAAP,iB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAAd,C;UAAyB,eAAO,O;UAAP,iB;;QACzB,eI/9CqB,QJ+9CN,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIi+CIB,QJk+CT,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QACIB,eAAO,O;;;MIx+CP,mB;IACJ,C;;FAEA,yB;MAAA,kD;MAAA,wC;QAKI,OAAe,QAAR,iBAAQ,EAAQ,UAAR,C;MACnB,C;KANA,C;;MAcmB,kBAAR,iB;MAAQ,gB;;QJkiDf,eAAe,sB;QACf,IAAI,CAAC,QAAS,UAAAd,C;UAAyB,eAAO,I;UAAP,iB;;QACzB,cAAc,QAAS,O;QACvB,IAAI,CAAC,QAAS,UAAAd,C;UAAyB,eAAO,O;UAAP,iB;;QACzB,eItdqB,QJsiDN,CAAS,OAAT,C;;UAEX,QAAQ,QAAS,O;UACjB,QIzidiB,QJyidT,CAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAED,QAAT,QAAS,W;QACIB,eAAO,O;;;MI/iDP,mB;IACJ,C;;MAMI,OAAe,UAAR,iBAAQ,EAAQ,UAAR,C;IACnB,C;;MAQI,OAAO,mB;IACX,C;;MASoB,Q;MADhB,IAAI,mBAAJ,C;QAAe,OAAO,I;MACN,OAAA,ShBuHoE,QAAQ,W;MgBvH5F,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,K;;MACrD,OAAO,I;IACX,C;;MAOmC,Q;MAAA,OhB8GqD,iBAAQ,W;MgB9G7E,OAAgB,cAAhB,C;QAAgB,yB;QAAM,OAAO,OAAp,C;;MAArC,gB;IACJ,C;;MAOI,OAAO,iB;IACX,C;;MAMI,OAAe,aAAR,iBAAQ,C;IACnB,C;gFtBjNA,yB;MAAA,mC;MAAA,2C;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;MACX,C;KATA,C;gFAWA,yB;MAAA,mC;MAAA,2C;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;MACX,C;KATA,C;;;QAmBQ,OAAc,QAAP,MAAO,EAAQ,SAAR,C;;;QACHb,+C;UACE,MAAM,2BAAuB,CAAE,QAazB,C;;;UAHV,O;;;IAKJ,C;;;QAUQ,OAAc,SAAP,MAAO,EAAS,SAAT,C;;;QACHb,+C;UACE,MAAM,2BAAuB,CAAE,QAazB,C;;;UAHV,O;;;IAKJ,C;;;QAUQ,OAAuD,OAAhD,MAAO,iBAAQ,eAAM,IAAd,EAAuB,CAAA,cAAK,IAAL,IAAe,CAAf,IAAvB,CAAYC,C;;;QACzD,+C;UACE,MAAM,2BAAuB,CAAE,QAazB,C;;;UAHV,O;;;IAKJ,C;;;MAUI,OAAO,mBAAmB,2BAAS,OAAT,C;IAC9B,C;;MAUI,OAAO,mBAAmB,2BAAS,OAAT,C;IAC9B,C;;MAUI,OAAO,2BAAE,KAAf,C;IACX,C;;MAOI,OAAO,2BAAE,oBAAN,KAAf,C;IACX,C;;MAOI,OAAO,2BAAE,KAAf,C;IACX,C;;MAQI,OAAO,2BAAE,KAAf,C;IACX,C;;MgBtDW,ShB8DM,mBAAN,KAAm,C;MAAb,OAA0C,UAAJ,GAAGB,2BAAS,EAAT,CAAhB,GAAC,K;IAC5E,C;;MgB/DW,ShBuEM,kBAAN,KAAm,C;MAAb,OAA2C,UAAJ,GAAGB,2BAAS,EAAT,CAAhB,GAAC,K;IAC7E,C;;MgBxEW,ShBgFM,oBAAN,KAAm,C;MAAb,OAA2C,UAAJ,GAAGB,2BAAS,EAAT,CAAhB,GAAC,K;IAC7E,C;;MgBjFW,ShByFM,qBAAN,KAAm,C;MAAb,OAA4C,UAAJ,GAAGB,2BAAS,EAAT,CAAhB,GAAC,K;IAC9E,C;;MAOI,OAAO,2BAAE,KAAf,C;IACX,C;;MgBIGW,ShB0GM,mBAAN,KAAm,C;MAAb,OAA0C,UAAJ,GAAGB,2BAAS,EAAT,CAAhB,GAAC,K;IAC5E,C;;MgB3GW,ShBmHM,oBAAN,KAAm,C;MAAb,OAA2C,UAAJ,GAAGB,2BAAS,EAAT,CAAhB,GAAC,K;IAC7E,C;;MgBpHW,ShB4HM,oBAAN,KAAm,C;MAAb,OAA2C,UAAJ,GAAGB,2BAAS,EAAT,CAAhB,GAAC,K;IAC7E,C;;MgB7HW,ShBqIM,qBAAN,KAAm,C;MAAb,OAA4C,UAAJ,GAAGB,2BAAS,EAAT,CAAhB,GAAC,K;IAC9E,C;;MAOI,OAAO,2BAAE,KAAf,C;IACX,C;;MAOI,OAAO,2BAAE,oBAAN,KAAm,CAAf,C;IACX,C;;MgBtJW,ShB6JM,kBAAN,KAAm,C;MAAb,OAA2C,UAAJ,GAAGB,2BAAS,EAAT,CAAhB,GAAC,K;IAC7E,C;;MgB9JW,ShBqKM,mBAAN,KAAm,C;MAAb,OAA4C,UAAJ,GAAGB,2BAAS,EAAT,C

AAhB,GAakC,K;IAC9E,C;;MAQI,OAAO,2BA Ae,KAAf,C;IACX,C;;MAQI,OAAO,2BA Ae,KAAf,C;IACX,C;;MgBxLW,ShB+LM,iBAAN,KAAM,C;MAAb,OAA0C,UAAJ,GAAgB,2BAAS,EAAT,CAAhB,GAakC,K;IAC5E,C;;MgBhMW,ShBuMM,oBAAN,KAAM,C;MAAb,OAA2C,UAAJ,GAAgB,2BAAS,EAAT,CAAhB,GAakC,K;IAC7E,C;;MgBxMW,ShB+MM,qBAAN,KAAM,C;MAAb,OAA4C,UAAJ,GAAgB,2BAAS,EAAT,CAAhB,GAakC,K;IAC9E,C;;MAQI,OAAO,2BAAS,KAAM,WAAf,C;IACX,C;;MAQI,OAAO,2BAAS,KAAM,WAAf,C;IACX,C;;MAOI,OAAO,2BA Ae,KAAf,C;IACX,C;;MAOI,OAAO,2BA Ae,oBAAN,KAAM,CAAf,C;IACX,C;;MgBIPW,ShByPM,oBAAN,KAAM,C;MAAb,OAA2C,UAAJ,GAAgB,2BAAS,EAAT,CAAhB,GAakC,K;IAC7E,C;;MAQI,OAAO,2BA Ae,KAAf,C;IACX,C;;MAQI,OAAO,2BA Ae,KAAf,C;IACX,C;;MASI,OAAO,sCA Ae,yBAAgB,SAAhB,EAAYB,EAzB,EAakC,EAaIC,C;IAC1B,C;;MASI,OAAO,uCAAgB,yBAAgB,SAAhB,EAAYB,oBAAH,EAAG,CAAzB,M;IAC3B,C;;MASI,OAAO,sCA Ae,yBAAgB,SAArB,EAaIC,EAaJC,EAa0C,EAa1C,C;IAC1B,C;;MASI,OAAO,sCA Ae,yBAAgB,SAArB,EAaIC,EAaJC,EAa0C,EAa1C,C;IAC1B,C;;MASI,OAAO,uCAAgB,yBAAgB,SAAhB,EAAsB,EAAtB,EAa0B,EAa1B,C;IAC3B,C;;MASI,OAAO,sCA Ae,yBAAgB,SAAhB,EAAsB,EAAtB,EAa0B,EAa1B,C;IAC1B,C;;MASI,OAAO,uCAAgB,yBAAgB,SAAhB,EAAYB,oBAAH,EAAG,CAAzB,M;IAC3B,C;;MASI,OAAO,sCA Ae,yBAAgB,SAArB,EAa8B,EAa9B,EAakC,EAaIC,C;IAC1B,C;;MASI,OAAO,sCA Ae,yBAAgB,SAArB,EAa8B,EAa9B,EAakC,EAaIC,C;IAC1B,C;;MASI,OAAO,uCAAgB,yBAAgB,oBAAL,SAAK,CAArB,EAa+B,EAa/B,M;IAC3B,C;;MASI,OAAO,uCAAgB,yBAAgB,SAAhB,EAAsB,EAAtB,M;IAC3B,C;;MASI,OAAO,uCAAgB,yBAAgB,oBAAL,SAAK,CAArB,EAa+B,EAa/B,M;IAC3B,C;;MASI,OAAO,sCA Ae,yBAAgB,SAAhB,EAAYB,EAzB,EAakC,EAaIC,C;IAC1B,C;;MASI,OAAO,uCAAgB,yBAAgB,SAAhB,EAAYB,oBAAH,EAAG,CAAzB,M;IAC3B,C;;MASI,OAAO,sCA Ae,yBAAgB,SAArB,EAaIC,EAaJC,EAa0C,EAa1C,C;IAC1B,C;;MAMI,OAAO,sCA Ae,yBAAgB,cAAhB,EAAsB,eAAtB,EAa6B,CAAC,cAAD,IAA7B,C;IAC1B,C;;MAMI,OAAO,uCAAgB,yBAAgB,cAAhB,EAAsB,eAAtB,EAa6B,CAAC,cAAD,IAA7B,C;IAC3B,C;;MAMI,oBAAoB,OAAO,CAA3B,EAa8B,IAA9B,C;MACA,OAAO,sCA Ae,yBAAgB,eAAhB,EAaUB,cAAvB,EAaIC,SAAK,KAAL,GAAY,CAAhB,GAAMB,IAANB,GAA6B,CAAC,IAAD,IAA1D,C;IAC1B,C;;MAMI,oBAAoB,kBAAO,CAA3B,EAa8B,IAA9B,C;MACA,OAAO,uCAAgB,yBAAgB,eAAhB,EAaUB,cAAvB,EAaIC,SAAK,KAAL,cAAy,CAAhB,GAAMB,IAANB,GAA6B,CAAC,IAAD,IAA1D,C;IAC3B,C;;MAMI,oBAAoB,OAAO,CAA3B,EAa8B,IAA9B,C;MACA,OAAO,uCAAgB,yBAAgB,eAAhB,EAaUB,cAAvB,EAaIC,SAAK,KAAL,GAAY,CAAhB,GAAMB,IAANB,GAA6B,CAAC,IAAD,IAA1D,C;IAC3B,C;;MAGI,OAAmB,IAAR,8BAAGC,GAApC,GAAiE,OAAL,SAAK,CAAjE,GAA+E,I;IAC1F,C;;MAGI,OAAW,mEAAJ,GAAmE,OAAL,SAAK,SAAnE,GAAiF,I;IAC5F,C;;MAGI,OAAW,YAAQ,aAAA,sCA Ae,UAAf,EAa0B,sCA Ae,UAAzC,CAAR,YAAJ,GAAqE,OAAL,SAAK,CAArE,GAAMF,I;IAC9F,C;;MAGI,OAAmB,UAAA,sCA Ae,UAAf,EAa2B,sCA Ae,UAA1C,CAAR,4BAAJ,GAAuE,mBAAL,SAAK,EAavE,GAAqF,I;IAC7G,C;;MAGI,OuB31BgB,UvB21BG,sCA Ae,UuB31BIB,EvB21B6B,sCA Ae,UuB31B5C,CvB21BL,4BAAJ,GAAqE,mBAAL,SAAK,EAArE,GAAMF,I;IAC9F,C;;MAGI,OAAW,iFAAJ,GAA4D,SAAK,QAAjE,GAA8E,I;IACzF,C;;MAGI,OAAmB,UAAc,WAAAd,EAaWc,UAAxC,CAAR,4BAAJ,GAAqE,YAAL,SAAK,CAArE,GAakF,I;IAC7F,C;;MAGI,OuBvmBgB,UvBumBiB,WuBvmBjB,EvBumB0C,UuBvmB1C,CvBumBL,4BAAJ,GAAmE,YAAL,SAAK,CAAnE,GAAgF,I;IAC3F,C;;MAGI,OAAmB,UAAe,mCAAf,EAa0C,mCAA1C,CAAR,4BAAJ,GAAuE,uBAAL,SAAK,CAAvE,GAAqF,I;IAC7G,C;;MAGI,OuB/mBgB,UvB+mBkB,mCuB/mBIB,EvB+mB4C,mCuB/mB5C,CvB+mBL,4BAAJ,GAAqE,uBAAL,SAAK,CAArE,GAAMF,I;IAC9F,C;;MAGI,OAAmB,MAAR,8BAaIC,KAArC,GAAMe,QAAL,SAAK,CAAnE,GAakF,I;IAC7F,C;;MAGI,OAAW,uEAAJ,GAAqE,QAAL,SAAK,SAArE,GAAoF,I;IAC/F,C;;MAGI,OAAmB,UAAA,uCAAgB,UAAhB,EAa4B,uCAAgB,UAA5C,CAAR,4BAAJ,GAAyE,oBAAL,SAAK,EAazE,GAAwF,I;IACnG,C;;MAGI,OuB/nBgB,UvB+nBG,uCAAgB,UuB/nBnB,EvB+nB8B,uCAAgB,UuB/nB9C,CvB+nBL,4BAAJ,GAAuE,oBAAL,SAAK,EAavE,GAAsF,I;IACjG,C;;MAQI,OAAO,wBAAY,EAaA,GAaH,CAAG,IAazB,C;IACX,C;;MAQI,OAAO,kBAAY,oBAAH,EAAG,CAAc,8BAAH,CAAG,EAa1B,C;IACX,C;;MAQI,OAAO,aAAK,SAAL,EAaOB,EAaA,GAaH,CAAG,IAAjC,C;IACX,C;;MAQI,OAAO,aAAK,SAAL,EAaOB,EAaA,GAaH,CAAG,IAAjC,C;IACX,C;;MAQI,IAAI,MAAM,CAAV,C;QAAoB,OAAO,iCAAU,M;MACrC,OAAO,yBAaIB,OAAR,EAaQ,GAaH,CAAG,CAAjB,C;IACX,C;;MAQI,IAAI,MAAM,WAAV,C;QAAyB,OAAO,gCAAS,M;MACzC,OAAO,wBAAS,EAaQ,GAaH,CAAG,IA

AjB,C;IACX,C;;MAQI,OAAO,kBAAY,oBAAH,EAAG,CAAc,8BAAH,CAAG,EAA1B,C;IACX,C;;MAQI,IAAI,MAAM,WAAV,C;QAAyB,OAAO,gCAAS,M;MACzC,OAAO,aAAK,SAAL,EAAiB,EAAQ,GAAH,CAAG,IAAzB,C;IACX,C;;MAQI,IAAI,MAAM,WAAV,C;QAAyB,OAAO,gCAAS,M;MACzC,OAAO,aAAK,SAAL,EAAiB,EAAQ,GAAH,CAAG,IAAzB,C;IACX,C;;MAQI,IAAI,iDAAJ,C;QAA0B,OAAO,iCAAU,M;MAC3C,OAAy,oBAA L,SAAK,CAAL,SAAkB,EAAQ,8BAAH,CAAG,EAA1B,C;IACX,C;;MAQI,IAAI,iDAAJ,C;QAA0B,OAAO,iCAA U,M;MAC3C,OAAO,kBAAS,EAAQ,8BAAH,CAAG,EAAjB,C;IACX,C;;MAQI,IAAI,iDAAJ,C;QAA0B,OAAO,i CAAU,M;MAC3C,OAAy,oBAA L,SAAK,CAAL,SAAkB,EAAQ,8BAAH,CAAG,EAA1B,C;IACX,C;;MAQI,IAAI,iDAAJ,C;QAA0B,OAAO,iCAAU,M;MAC3C,OAAy,oBAA L,SAAK,CAAL,SAAkB,EAAQ,8BAAH,CAAG,EAA1B,C;IACX,C;;MAQI,IAAI,iDAAJ,C;QAA0B,OAAO,iCAAU,M;MAC3C,OAAy,oBAA L,SAAK,CAAL,SAAkB,EAAQ,8BAAH,CAAG,EAA1B,C;IACX,C;;MAQI,OAAO,wBAAY,EAAa,GAAH,CAAG,IAAzB,C;IACX,C;;MAQI,OAAO,kBAAY,oBAA H,EAAG,CAAc,8BAAH,CAAG,EAA1B,C;IACX,C;;MAQI,OAAO,aAAK,SAAL,EAAoB,EAAa,GAAH,CAAG,IAA jC,C;IACX,C;;MAQI,OAAO,aAAK,SAAL,EAAoB,EAAa,GAAH,CAAG,IAA jC,C;IACX,C;;MAUI,OAAW,4 BAAO,YAAP,KAAJ,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAUI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA 2C,S;IACtD,C;;MAUI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAUI,OAAW,YAAO,YAA X,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAUI,OAAW,0BAAO,YAAP,KAAJ,GAAyB,YAAzB,GAA2C,S;IACtD ,C;;MAUI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAUI,OAAW,YAAO,YAAX,GAAyB,Y AAzB,GAA2C,S;IACtD,C;;MAUI,OAAW,4BAAO,YAAP,KAAJ,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAUI,O AAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAUI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA 2C,S;IACtD,C;;MAUI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAUI,OAAW,0BAAO,YAA P,KAAJ,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAUI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;IACtD, C;;MAUI,OAAW,YAAO,YAAX,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAUI,IAAI,iBAAiB,IAAjB,IAAyB,iBA AiB,IAA9C,C;QACI,IAAI,+BAAe,YAAf,KAAJ,C;UAAiC,MAAM,gCAAyB,6DAAiD,YAAjD,wCAAoF,YAApF, OAAzB,C;QACvC,IAAI,4BAAO,YAAP,KAAJ,C;UAAyB,OAAO,Y;QAChC,IAAI,4BAAO,YAAP,KAAJ,C;UAA yB,OAAO,Y;;;QAGhC,IAAI,iBAAiB,IAAjB,IAAyB,4BAAO,YAAP,KAA7B,C;UAAkD,OAAO,Y;QACzD,IAAI, iBAAiB,IAAjB,IAAyB,4BAAO,YAAP,KAA7B,C;UAAkD,OAAO,Y;;MAE7D,OAAO,S;IACX,C;;MAUI,IAAI,e AAe,YAA nB,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IAAI,YAAO, YAAX,C;QAAyB,OAAO,Y;MACHC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MACHC,OAAO,S;IACX,C;;MAU I,IAAI,eAAe,YAA nB,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IAAI, YAAO,YAAX,C;QAAyB,OAAO,Y;MACHC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MACHC,OAAO,S;IACX, C;;MAUI,IAAI,eAAe,YAA nB,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACv C,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MACHC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MACHC,OAAO,S ;IACX,C;;MAUI,IAAI,6BAAe,YAAf,KAAJ,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,yCAAoF,YAApF,iBA AzB,C;MACvC,IAAI,0BAAO,YAAP,KAAJ,C;QAAyB,OAAO,Y;MACHC,IAAI,0BAAO,YAAP,KAAJ,C;QAAy B,OAAO,Y;MACHC,OAAO,S;IACX,C;;MAUI,IAAI,eAAe,YAA nB,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAj D,8BAAoF,YAApF,MAAzB,C;MACvC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MACHC,IAAI,YAAO,YAAX, C;QAAyB,OAAO,Y;MACHC,OAAO,S;IACX,C;;MAUI,IAAI,eAAe,YAA nB,C;QAAiC,MAAM,gCAAyB,oDAAi D,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IAAI,YAAO,YAAX,C;QAAyB,OAAO,Y;MACHC,IAAI,YAAO, YAAX,C;QAAyB,OAAO,Y;MACHC,OAAO,S;IACX,C;;MAYW,Q;MADP,IAAI,KAAM,UAAV,C;QAAqB,MAA M,gCAAyB,4CAAyC,KAAzC,MAAzB,C;MAGvB,IAAA,KAAM,0BAAiB,SAAjB,EAAuB,KAAM,MAA7B,CA AN,IAA6C,CAAC,KAAM,0BAAiB,KAAM,MAAvB,EAA8B,SAA9B,CAApD,C;QAAiG,OAAN,KAAM,M;WA EjG,IAAA,KAAM,0BAAiB,KAAM,aAAvB,EAAqC,SAArC,CAAN,IAAoD,CAAC,KAAM,0BAAiB,SAAjB,EAA uB,KAAM,aAA7B,CAA3D,C;QAA+G,OAAN,KAAM,a;;QACvG,gB;MALZ,W;IAOJ,C;;MAcW,Q;MAJP,IAAI,8 CAAJ,C;QACI,OAAy,WAAL,SAAK,EAAY,KAAZ,C;;MAEhB,IAAI,KAAM,UAAV,C;QAAqB,MAAM,gCAAy B,4CAAyC,KAAzC,MAAzB,C;MAEvB,gCAAO,KAAM,MAAb,M;QAA4B,OAAN,KAAM,M;WAC5B,gCAAO, KAAM,aAAb,M;QAAmC,OAAN,KAAM,a;;QAC3B,gB;MAHZ,W;IAKJ,C;;MAcW,Q;MAJP,IAAI,8CAAJ,C;QA CI,OAAy,WAAL,SAAK,EAAC,KAA d,C;;MAEhB,IAAI,KAAM,UAAV,C;QAAqB,MAAM,gCAAyB,4CAAyC,K AAzC,MAAzB,C;MAEvB,gBAAO,KAAM,MAAb,C;QAA4B,OAAN,KAAM,M;WAC5B,gBAAO,KAAM,aAAb, C;QAAmC,OAAN,KAAM,a;;QAC3B,gB;MAHZ,W;IAKJ,C;;MAcW,Q;MAJP,IAAI,8CAAJ,C;QACI,OAAy,WA AL,SAAK,EA Ae,KAAf,C;;MAEhB,IAAI,KAAM,UAAV,C;QAAqB,MAAM,gCAAyB,4CAAyC,KAAzC,MAAzB

,C;MAEvB,8BAAO,KAAM,MAAb,M;QAA4B,OAAN,KAAM,M;WAC5B,8BAAO,KAAM,aAAb,M;QAAmC,O
AAN,KAAM,a;;QAC3B,gB;MAHZ,W;IAKJ,C;;MU1mCuF,wC;IANvF,C;IAAA,0D;MAO2C,8B;IAAA,C;IAP3C,
gF;;MccI,OAAO,sBAAQ,OAAR,KAAoB,C;IAC/B,C;;MAUkC,qB;QAAE,MAAM,8BAA0B,+CAA4C,aAA5C,M
AA1B,C;MAAR,C;IAAA,C;;MAA9B,OAAO,8BAAgB,KAAbB,EAABuB,yBAAvB,C;IACX,C;;MAGbqB,Q;MANj
B,IAAI,QAAQ,CAAZ,C;QACI,OAAO,aAAa,KAAb,C;MACX,eAAe,oB;MACf,YAAY,C;MACZ,OAAO,QAAS,U
AAhB,C;QACI,cAAc,QAAS,O;QACvB,IAAI,WAAS,YAAT,EAAS,oBAAT,OAAJ,C;UACI,OAAO,O;;MAEf,OA
AO,aAAa,KAAb,C;IACX,C;;MAGbqB,Q;MANjB,IAAI,QAAQ,CAAZ,C;QACI,OAAO,I;MACX,eAAe,oB;MACf,
YAAY,C;MACZ,OAAO,QAAS,UAAhB,C;QACI,cAAc,QAAS,O;QACvB,IAAI,WAAS,YAAT,EAAS,oBAAT,O
AAJ,C;UACI,OAAO,O;;MAEf,OAAO,I;IACX,C;;MASW,sB;;QAUdS,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;U
AAgB,yB;UAAM,IAvDH,SAuDO,CAAU,OAAV,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;QAC9C,qBAAO,I;;M
AxDP,yB;IACJ,C;;MASLoB,Q;MADhB,WAAe,I;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IA9K
c,SA8KV,CAAU,OAAV,CAAJ,C;UACI,OAAO,O;;MA/Kf,OkLO,I;IAjLX,C;;MASI,eAAe,oB;MACf,IAAI,CA
AC,QAAS,UAAc,C;QACI,MAAM,2BAAuB,oBAAvB,C;MACV,OAAO,QAAS,O;IACpB,C;iFAEA,yB;MAAA,i
E;MAAA,uC;QAOoB,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,UAAU,OAAV,CAAJ,C
;YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,sDAAvB,C;MACV,C;KATA,C;;MAiBI,eAAe,oB;MACf,IAAI,CA
AC,QAAS,UAAc,C;QACI,OAAO,I;MACX,OAAO,QAAS,O;IACpB,C;;MAQoB,Q;MAAA,2B;MAAhB,OAAgB,
cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;;MAS
iB,Q;MADb,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,mBAAmB,KAAAnB,C;QACA,IAAI,gB
AAW,IAAX,CAAJ,C;UACI,OAAO,K;QACX,qB;;MAEJ,OAAO,E;IACX,C;+FAEA,yB;MAAA,wE;MAAA,uC;Q
AOiB,Q;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,mBAAmB,KAAAnB,C;UACA,IAAI,
UAAU,IAAV,CAAJ,C;YACI,OAAO,K;UACX,qB;;QAEJ,OAAO,E;MACX,C;KAdA,C;6FAgBA,yB;MAAA,wE;
MAAA,uC;QAQiB,Q;QAFb,gBAAgB,E;QChB,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,m
BAAmB,KAAAnB,C;UACA,IAAI,UAAU,IAAV,CAAJ,C;YACI,YAAY,K;UChB,qB;;QAEJ,OAAO,S;MACX,C;
KafA,C;;MAwBI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAc,C;QACI,MAAM,2BAAuB,oBAAvB,C;MACV,WA
AW,QAAS,O;MACpB,OAAO,QAAS,UAAhB,C;QACI,OAAO,QAAS,O;MACpB,OAAO,I;IACX,C;+EAEA,yB;
MAAA,iE;MAAA,gB;MAAA,8B;MAAA,uC;QASoB,UAQT,M;QAVP,WAAe,I;QACf,YAAY,K;QACI,2B;QAAh
B,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,OAAV,CAAJ,C;YACI,OAAO,O;YACP,QAAQ,I;;QAGhB,
IAAI,CAAC,KAAL,C;UAAy,MAAM,gCAAuB,sDAAvB,C;QAEIB,OAAO,2E;MACX,C;KAIBA,C;;MA4BiB,Q;
MAFb,gBAAgB,E;MACHB,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,mBAAmB,KAAAnB,C;Q
ACA,IAAI,gBAAW,IAAX,CAAJ,C;UACI,YAAY,K;QChB,qB;;MAEJ,OAAO,S;IACX,C;;MAQI,eAAe,oB;MA
Cf,IAAI,CAAC,QAAS,UAAc,C;QACI,OAAO,I;MACX,WAAW,QAAS,O;MACpB,OAAO,QAAS,UAAhB,C;QA
CI,OAAO,QAAS,O;MACpB,OAAO,I;IACX,C;;MASoB,Q;MADhB,WAAe,I;MACC,2B;MAAhB,OAAgB,cAAhB
,C;QAAgB,yB;QACZ,IAAI,UAAU,OAAV,CAAJ,C;UACI,OAAO,O;;MAGf,OAAO,I;IACX,C;;MAQI,eAAe,oB;
MACf,IAAI,CAAC,QAAS,UAAc,C;QACI,MAAM,2BAAuB,oBAAvB,C;MACV,aAAa,QAAS,O;MACTB,IAAI,Q
AAS,UAAb,C;QACI,MAAM,gCAAyB,qCAAzB,C;MACV,OAAO,M;IACX,C;mFAEA,yB;MAAA,kF;MAAA,iE;
MAAA,gB;MAAA,8B;MAAA,uC;QAQoB,UAST,M;QAXP,aAAiB,I;QACjB,YAAY,K;QACI,2B;QAAhB,OAAg
B,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,mD
AAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;QAGhB,IAAI,CAAC,KAAL,C;UAAy,MAAM,gCAAuB,sDAAvB,C;
QAEIB,OAAO,6E;MACX,C;KAIBA,C;;MA0BI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAc,C;QACI,OAAO,I;M
ACX,aAAa,QAAS,O;MACTB,IAAI,QAAS,UAAb,C;QACI,OAAO,I;MACX,OAAO,M;IACX,C;;MAUoB,Q;MAF
hB,aAAiB,I;MACjB,YAAY,K;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,UAAU,OAAV,CA
AJ,C;UACI,IAAI,KAAJ,C;YAAW,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;MAGhB,IAAI,CAAC,KAAL,C;Q
AAY,OAAO,I;MACnB,OAAO,M;IACX,C;;MAAw,Q;MZhUP,IAAI,EY+TI,KAAK,CZ/TT,CAAJ,C;QACI,cY8Tc
,sD;QZ7Td,MAAM,gCAAyB,OAAQ,WAAjC,C;;MY+TN,UAAK,CAAL,C;QAAU,gB;WACV,+C;QAAiC,OAAL
,SAAK,cAAK,CAAL,C;;QACzB,wBAAa,SAAb,EAAMB,CAAnB,C;MAHZ,W;IAKJ,C;;MAUI,OAAO,sBAAkB,
SAAIB,EAAwB,SAAXB,C;IACX,C;;MAQI,OAAO,sBAAkB,SAAIB,EAAwB,IAAXB,EAA8B,SAA9B,C;IACX,C;
;MAWgF,qB;QAAE,yBAAU,EAAG,MAAb,EAAoB,EAAG,MAAvB,C;MAAF,C;IAAA,C;;MAAsC,OAAA,EAA
G,M;IAAM,C;;MAA3H,OAAO,yBAAqB,sBAAkB,qBAAiB,SAAjB,CAAIb,EAA0C,IAA1C,EAAGD,+BAAhD,C

AArB,EAAYG,sBAAZG,C;IACX,C;oGAEA,yB;MAorBA,wE;MAprBA,oD;QA6rBiB,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UAprBT,IAAI,UAorBkB,oBAAmB,cAAnB,EAAMb,sBAAnB,UAprBIB,EAo rB+C,IAprB/C,CAAJ,C;YAA2C,sBAorBQ,IAprBR,C;;QAE/C,OAAO,W;MACX,C;KAZA,C;sGAcA,yB;MAAA,8 C;MAAA,0C;MAAA,8B;;QAOKb,qB;UAAE,c;QAAF,C;MAAA,C;MAPIB,sC;QAOW,Q;QAAP,OAAO,uCAAO,i CAAP,gC;MACX,C;KARA,C;;MAgBoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,YA AJ,C;UAAkB,WAAY,WAAI,OAAJ,C;;MACpD,OAAO,W;IACX,C;;MAQI,OAAO,sBAAkB,SAaIB,EAAwB,KA AxB,EAA+B,SAA/B,C;IACX,C;;MASuB,iB;IAAW,C;;MAAvB,Q;MAAP,OAAO,4CAAU,oBAAV,kC;IACX,C;; MAQoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,eAAJ,C;UAAqB,WAAY,WAAI,OA AJ,C;;MACvD,OAAO,W;IACX,C;;MAQoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI, CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;IACX,C;;MAQoB,Q;MAA A,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,WAAY,WAAI,OA AJ,C;;MAC1D,OAAO,W;IACX,C;;MAAw,Q;MZrcP,IAAI,EYocI,KAAK,CZpcT,CAAJ,C;QACI,cYmcc,sD;QZlcl d,MAAM,gCAAYB,OAAQ,WAAjC,C;;MYocN,UAAK,CAAL,C;QAAU,sB;WACV,+C;QAAiC,OAAL,SAAK,cA AK,CAAL,C;;QACzB,wBAaA,SAAb,EAAMb,CAAnB,C;MAHZ,W;IAKJ,C;;MAUI,OAAO,sBAAkB,SAaIB,EA AwB,SAAXB,C;IACX,C;;MAUW,8B;IAAA,C;;MAEC,iBAA6B,iBAAZ,gBAAy,C;MACiB,QAAX,UAAW,C;M ACX,OAAO,UAAW,W;IACtB,C;;MALJ,0C;IAOJ,C;sFAEA,yB;MAAA,sD;MV7aA,sC;MCjFA,kC;MAAA,oC;; QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDiFA,uBAO e,yB;QArEf,8D;;UAqEe,uB;YAAU,eAAsB,gB;YAAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2B,SA4DM, CA5DN,CAA3B,C;UA4DI,C;QAAA,C;OAAA,C;MUsaf,sC;QAQI,OAAO,mDV9aI,iBU8aiB,QV9ajB,CU8aj,E;M ACX,C;KATA,C;0GAWA,yB;MAAA,sD;MVraA,sC;MCpGA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D; QAC4C,+BAAW,CAAX,EAAC,CAAd,C;MAAA,C;MAD5C,kF;MDoGA,iCAOe,yB;QAxFf,8D;;UAwFe,uB;YAA U,eAAsB,gB;YAAAtB,OA/Ed,cAAc,SA+EgB,CA/EhB,CAAd,EAA2B,SA+EM,CA/EN,CAA3B,C;UA+EI,C;QAA A,C;OAAA,C;MU8Zf,sC;QAQI,OAAO,mDVtaI,2BUsa2B,QVta3B,CUsaJ,E;MACX,C;KATA,C;;MAmBI,OAAO, wBAAW,cAAX,C;IACX,C;;MAUW,sC;MAAA,4C;IAAA,C;;MAEC,iBAAiC,iBAAhB,oBAAgB,C;MACtB,WAA X,UAAW,EAAS,uBAAT,C;MACX,OAAO,UAAW,W;IACtB,C;;MALJ,0D;IAOJ,C;wFAEA,yB;MAAA,wE;MA AA,uC;QAWW,kBAAY,oB;QAuEH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,WAXEsC,SAw EvB,CAAU,OAAV,C;UIBrCnB,wBAALI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;QkBnCA,OA0EO,W;MAzEX,C ;KAZA,C;6FAcA,yB;MAAA,wE;MAAA,yC;QAWW,kBAAc,oB;QA0BL,Q;QAAA,2B;QAAhB,OAAgB,cAAhB, C;UAAgB,yB;UACZ,WAAY,aA3B4B,WA2BxB,CAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;QA3BhB,OA6BO,W; MA5BX,C;KAZA,C;6FAcA,yB;MAAA,wE;MAAA,yD;QAUW,kBAAc,oB;QA6BL,Q;QAAA,2B;QAAhB,OAAg B,cAAhB,C;UAAgB,yB;UACZ,WAAY,aA9B4B,WA8BxB,CAAY,OAAZ,CAAJ,EA9ByC,cA8Bf,CAAe,OAaf,C AA1B,C;;QA9BhB,OA9CO,W;MA/BX,C;KAXA,C;;MAuBoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB ,yB;QACZ,WAAY,aAAI,YAAY,OAAZ,CAAJ,EAA0B,OAA1B,C;;MAEhB,OAAO,W;IACX,C;;MAYoB,Q;MAA A,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QACZ,WAAY,aAAI,YAAY,OAAZ,CAAJ,EAA0B,eAAe,OAaf,C AA1B,C;;MAEhB,OAAO,W;IACX,C;;MAWob,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QACZ,W AAe,UAAU,OAAV,C;QlBrCnB,wBAALI,IAAK,MAAT,EAAGB,IAAK,OAArB,C;;MkBuCA,OAAO,W;IACX,C;g GAEA,yB;MAAA,wE;MAAA,2C;QAcI,aAAa,oB;QAcG,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;U AbO,MAcP,aAAI,OAAJ,EAd,e,aAcF,CAAc,OAAc,CAAb,C;;QadhB,OAAuB,M;MAC3B,C;KAhBA,C;;MA4BoB ,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QACZ,WAAY,aAAI,OAAJ,EAAa,cAAc,OAAc,CAAb,C;; MAEhB,OAAO,W;IACX,C;;MAQiB,Q;MAAA,2B;MAAb,OAAa,cAAb,C;QAAA,sB;QACT,WAAY,WAAI,IAAJ, C;;MAEhB,OAAO,W;IACX,C;;MAQI,OAAO,0BAAa,cAAb,C;IACX,C;;MAQI,OAA4B,qBAAhB,iBAAL,SAAK, CAAGB,C;IAChC,C;;MAQI,OAAO,0BAAa,gBAAb,C;IACX,C;;MAUI,OAAwC,oBAAjC,0BAAa,sBAAb,CAAiC ,C;IAC5C,C;;MAQiD,OAAA,EAAG,W;IAAW,C;;MAA3D,OAAO,uBAAmB,SAAnB,EAAYB,SAAZB,EAAoC,c AApC,C;IACX,C;yFAEA,yB;MAAA,gD;MAAA,oD;QAMoB,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,y B;UACZ,WAAY,UAAU,OAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KAXA, C;qFAaA,yB;MAAA,wE;MA6BA,+D;MA7BA,yC;QAWW,kBAAU,oB;QA6BD,Q;QAAA,2B;QAAhB,OAAgB,c AAhB,C;UAAgB,yB;UACZ,UA9BiD,WA8BvC,CAAY,OAAZ,C;UIBvfP,U;UADP,YkByfe,WIBzfH,WkByfwB,G lBzfxB,C;UACL,IAAI,aAAJ,C;YACH,akBufuC,gB;YAA5B,WIBtfx,akBsfG,C;GIBtfc,EAAS,MAAT,C;YACA,e;

::YAEA,c;;UkBmfA,iB;UACA,IAAK,WAAI,OAAJ,C;;QAhCT,OAKCO,W;MAjCX,C;KAZA,C;qFAcA,yB;MAA
A,wE;MAkCA,+D;MAICA,yD;QAYW,kBAAU,oB;QAKCD,Q;QAAA,2B;QAaHb,OAAGb,cAAhB,C;UAAgB,yB
;UACZ,UAnCiD,WAmCvC,CAAY,OAAZ,C;UIB3gBP,U;UADP,YkB6gBe,WIB7gBH,WkB6gBwB,GIB7gBxB,C;
UACL,IAAI,aAAJ,C;YACH,akB2gBuC,gB;YAA5B,WIB1gBX,akB0gBgC,GIB1gBhC,EAAS,MAAT,C;YACA,e;;
;YAEA,c;;UkBugBA,iB;UACA,IAAK,WArCyD,cAqCrD,CAAe,OAAf,CAAJ,C;;QArCT,OAuCO,W;MAtCX,C;K
AbA,C;yFAeA,yB;MAAA,+D;MAAA,sD;QAWoB,Q;QAAA,2B;QAaHb,OAAGb,cAAhB,C;UAAgB,yB;UACZ,
UAAU,YAAY,OAAZ,C;UIBvfP,U;UADP,YkByfe,WIBzfH,WkByfwB,GIBzfxB,C;UACL,IAAI,aAAJ,C;YACH,a
kBufuC,gB;YAA5B,WIBtfx,akBsfG,C,GIBtfc,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UkBmfA,iB;UACA,IAAK,W
AAI,OAAJ,C;;QAET,OAAO,W;MACX,C;KAjBA,C;yFAmBA,yB;MAAA,+D;MAAA,sE;QAYoB,Q;QAAA,2B;
QAaHb,OAAGb,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAAY,OAAZ,C;UIB3gBP,U;UADP,YkB6gBe,WIB7gBH,
WkB6gBwB,GIB7gBxB,C;UACL,IAAI,aAAJ,C;YACH,akB2gBuC,gB;YAA5B,WIB1gBX,akB0gBgC,GIB1gBhC,
EAAS,MAAT,C;YACA,e;;;YAEA,c;;UkBugBA,iB;UACA,IAAK,WAAI,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;
MACX,C;KAIBA,C;OFAoBA,yB;MAAA,kC;MAAA,4C;;QAUW,sC;QAAA,8C;MAVX,C;MAAA,gE;QAWqD,O
AAA,oBAAGB,W;MAAhB,C;MAXrD,oE;QAY4C,gCAAY,OAAZ,C;MAAA,C;MAZ5C,gF;MAAA,yC;QAUI,2D;
MAIJ,C;KAdA,C;;MAyBI,OAAO,yBAAqB,SAArB,EAA2B,SAA3B,C;IACX,C;;MAWI,OAAO,gCAA4B,SAA5B
,EAAkC,SAAIC,C;IACX,C;;MAWI,OAAoD,gBAA7C,gCAA4B,SAA5B,EAAkC,SAAIC,CAA6C,C;IACxD,C;4G
AEA,yB;MAuMA,wE;MAvMA,oD;QAgNiB,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UA
vMsB,U;UAAA,wBAuMT,oBAAmB,cAAnB,EAAMb,sBAAnB,UAvMS,EAuMoB,IAvMpB,W;YAA6C,6B;;;QA
ChF,OAAO,W;MACX,C;KAXA,C;8FAaA,yB;MAAA,wE;MAAA,oD;QAUIB,UACoC,M;QAFjD,YAAY,C;QAC
C,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAAY,WAAI,UAAU,oBAAmB,cAAnB,EAAMb,sBAAnB,UAAV,
EAAuC,IAAvC,CAAJ,C;;QACHB,OAAO,W;MACX,C;KAbA,C;;MASBI,OAA6C,gBAATC,yBAAqB,SAArB,EA
A2B,SAA3B,CAAsC,C;IACjD,C;8FAEA,yB;MAAA,oD;QA8JoB,Q;QAAA,2B;QAaHb,OAAGb,cAAhB,C;UAA
gB,yB;UAvJK,U;UAAA,wBAuJQ,OAvJR,W;YAAsC,6B;;;QAC3D,OAAO,W;MACX,C;KATA,C;;MAkBiB,Q;M
AAA,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,WAAY,WAAI,UAAU,IAAV,CAAJ,C;;MACHB,OAAO,W;IAC
X,C;;MASI,OAAO,qBAAiB,SAAjB,C;IACX,C;;MAU6B,S;IAAG,C;;MAA5B,OAAy,aAAL,SAAK,EAAW,eAA
X,C;IACHB,C;;MAWI,OAAO,qBAAiB,SAAjB,EAAuB,QAAvB,C;IACX,C;;MAWiB,Q;MADb,UAAU,sB;MACG
,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QAAM,GAAl,WAAI,IAAJ,C;;MACvB,OAAO,G;IACX,C;;MAUoB,Q;MA
AA,2B;MAAhB,OAAGb,cAAhB,C;QAAGb,yB;QAAM,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,OAAO,K;;
MACtD,OAAO,I;IACX,C;;MAUI,OAAO,oBAAW,U;IACtB,C;;MAUoB,Q;MAAA,2B;MAAhB,OAAGb,cAAhB,
C;QAAGb,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,I;;MACrD,OAAO,K;IACX,C;;MASoB,Q;M
ADhB,YAAY,C;MACI,2B;MAAhB,OAAGb,cAAhB,C;QAAGb,yB;QAAM,oBAAmB,qBAAnB,EAAMb,KAAAnB
,E;;MACtB,OAAO,K;IACX,C;FAEA,yB;MAAA,wE;MAAA,uC;QAOoB,Q;QADhB,YAAY,C;QACI,2B;QAaHb
,OAAGb,cAAhB,C;UAAgB,yB;UAAM,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,oBAAmB,qBAAnB,EAAMb,KA
AnB,E;;QAC9C,OAAO,K;MACX,C;KATA,C;;MAkBoB,Q;MADhB,kBAakB,O;MACF,2B;MAAhB,OAAGb,cA
AhB,C;QAAGb,yB;QAAM,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;IACX,C;4FAEA,yB;M
AAA,wE;MAAA,gD;QAWoB,UAAiD,M;QAFjE,YAAY,C;QACZ,kBAakB,O;QACF,2B;QAaHb,OAAGb,cAAh
B,C;UAAgB,yB;UAAM,cAAc,UAAU,oBAAmB,cAAnB,EAAMb,sBAAnB,UAAV,EAAuC,WAAvC,EAAoD,OA
ApD,C;;QACpC,OAAO,W;MACX,C;KAbA,C;;MAqBoB,Q;MAAA,2B;MAAhB,OAAGb,cAAhB,C;QAAGb,yB;
QAAM,OAAO,OAAP,C;;IAC1B,C;kGAEA,yB;MAAA,wE;MAAA,oC;QASiB,UAAgC,M;QAD7C,YAAY,C;QA
CC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UAAM,OAAO,oBAAmB,cAAnB,EAAMb,sBAAnB,UAAP,EAAoC,IAA
pC,C;;MACvB,C;KAVA,C;;MAqBI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAAd,C;QAAyB,OAAO,I;MACHC,UA
AU,QAAS,O;MACnB,IAAQ,QAAl,GAAl,CAAR,C;QAAiB,OAAO,G;MACxB,OAAO,QAAS,UAAhB,C;QACI,
QAAQ,QAAS,O;QACjB,IAAM,QAAl,CAAE,CAAN,C;UAAe,OAAO,C;QACtB,IAAI,MAAM,CAAV,C;UAAa,
MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAWI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAAd,C;QAAyB,OAAO,I;
MACHC,UAAU,QAAS,O;MACnB,IAAQ,QAAl,GAAl,CAAR,C;QAAiB,OAAO,G;MACxB,OAAO,QAAS,UAAh
B,C;QACI,QAAQ,QAAS,O;QACjB,IAAM,QAAl,CAAE,CAAN,C;UAAe,OAAO,C;QACtB,IAAI,MAAM,CAAV
,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAAAd,C;QAAyB,O
AAO,I;MACHC,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAI,sBAA

M,CAAN,KAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAUI,eAAe,oB;MACf,IAAI,CAAC,QAAS,U
AAd,C;QAAyB,OAAO,I;MACHc,cAAc,QAAS,O;MACvB,IAAI,CAAC,QAAS,UAd,C;QAAyB,OAAO,O;MAC
hC,eAAe,SAAS,OAAT,C;;QAEX,QAAQ,QAAS,O;QACjB,QAAQ,SAAS,CAAT,C;QACR,IAAI,2BAAW,CAAX,
KAAJ,C;UACI,UAAU,C;UACV,WAAW,C;;MAED,QAAT,QAAS,W;MACIB,OAAO,O;IACX,C;;MAQI,eAAe,o
B;MACf,IAAI,CAAC,QAAS,UAd,C;QAAyB,OAAO,I;MACHc,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAh
B,C;QACI,QAAQ,QAAS,O;QACjB,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC
,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAWI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAd,C;QAAyB,OAAO,I;
MACHc,UAAU,QAAS,O;MACnB,IAAQ,QAAJ,GAAL,CAAR,C;QAAiB,OAAO,G;MACxB,OAAO,QAAS,UAAh
B,C;QACI,QAAQ,QAAS,O;QACjB,IAAM,QAAF,CAAE,CAAN,C;UAAe,OAAO,C;QACtB,IAAI,MAAM,CAAV
,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAWI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAd,C;QAAyB,
OAAO,I;MACHc,UAAU,QAAS,O;MACnB,IAAQ,QAAJ,GAAL,CAAR,C;QAAiB,OAAO,G;MACxB,OAAO,QA
AS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAM,QAAF,CAAE,CAAN,C;UAAe,OAAO,C;QACtB,IAAI,MA
AM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAQI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAd,
C;QAAyB,OAAO,I;MACHc,UAAU,QAAS,O;MACnB,OAAO,QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,
IAAI,sBAAM,CAAN,KAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAUI,eAAe,oB;MACf,IAAI,CAA
C,QAAS,UAd,C;QAAyB,OAAO,I;MACHc,cAAc,QAAS,O;MACvB,IAAI,CAAC,QAAS,UAd,C;QAAyB,OA
AO,O;MACHc,eAAe,SAAS,OAAT,C;;QAEX,QAAQ,QAAS,O;QACjB,QAAQ,SAAS,CAAT,C;QACR,IAAI,2BA
AW,CAAX,KAAJ,C;UACI,UAAU,C;UACV,WAAW,C;;MAED,QAAT,QAAS,W;MACIB,OAAO,O;IACX,C;;M
AQI,eAAe,oB;MACf,IAAI,CAAC,QAAS,UAd,C;QAAyB,OAAO,I;MACHc,UAAU,QAAS,O;MACnB,OAAO,
QAAS,UAAhB,C;QACI,QAAQ,QAAS,O;QACjB,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAA
jC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAUI,OAAO,CAAC,oBAAW,U;IACvB,C;;MAUoB,Q;MA
AA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,K;;MACr
D,OAAO,I;IACX,C;;MASE,qB;QACP,eAAO,EAAP,C;QAAA,OACA,E;MAFO,C;IAAA,C;;MAAX,OAAO,kBAA
I,qBAAJ,C;IAIX,C;kFAEA,yB;MAAA,4F;MAAA,uC;QAMI,eAAe,SAAK,W;QACpB,IAAI,CAAC,QAAS,UAd,
C;UAAyB,MAAM,mCAA8B,kCAA9B,C;QAC/B,kBAAqB,QAAS,O;QAC9B,OAAO,QAAS,UAAhB,C;UACI,cAA
Ac,UAAU,WAAV,EAAuB,QAAS,OAAhC,C;;QAEIB,OAAO,W;MACX,C;KAbA,C;gGAeA,yB;MAAA,4F;MAA
A,wE;MAAA,uC;QAcM,D,Q;QAL/C,eAAe,SAAK,W;QACpB,IAAI,CAAC,QAAS,UAd,C;UAAyB,MAAM,mC
AA8B,kCAA9B,C;QAC/B,YAAy,C;QACZ,kBAAqB,QAAS,O;QAC9B,OAAO,QAAS,UAAhB,C;UACI,cAAc,U
AAU,oBAAmB,YAAnB,EAAmB,oBAAnB,QAAV,EAAuC,WAAvC,EAAoD,QAAS,OAA7D,C;;QAEIB,OAAO,
W;MACX,C;KAjBA,C;;MA0BoB,Q;MADhB,UAAe,C;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ
,YAAO,SAAS,OAAT,CAAP,I;;MAEJ,OAAO,G;IACX,C;;MASoB,Q;MADhB,UAAkB,G;MACF,2B;MAAhB,OA
AgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;IACX,C;;MAQe,qB;QAAE,e;UAA
M,MAAM,gCAAyB,2BAAwB,mBAAXB,MAAZB,C;;QAAZ,S;MAAF,C;IAAA,C;;MAAX,OAAO,kBAAI,gCAAJ
,C;IACX,C;;MAeI,OAAO,sBAAS,IAAT,EAAe,IAAf,EAAc,IAAtC,C;IACX,C;;MAoBI,OAAO,sBAAS,IAAT,E
AAe,IAAf,EAAc,IAAtC,EAAwD,SAAXD,C;IACX,C;;MAQW,4B;MAAA,sC;IAAA,C;;MAG0B,qB;QAAE,IAA
I,CAAC,iBAAD,IAAY,WAAM,eAAN,CAAhB,C;UAAiC,oBAAU,I;UAA3C,OAAiD,K;;UAAjD,OAA8D,I;MAA
hE,C;IAAA,C;;MADzB,kBAAc,KAAc,C;MACA,OAAkB,SAAX,eAAW,EAAO,kEAAP,CAA8E,W;IACpG,C;;M
AJJ,kD;IAMJ,C;;MAeW,wC;MAAA,4B;IAAA,C;;MAG6B,qB;QAAE,OAAM,aAAN,mB;MAAF,C;IAAA,C;;MA
D5B,YAAqB,UAAT,qBAAS,C;MACrB,OAAkB,YAAX,eAAW,EAAU,4CAAV,CAA0B,W;IACd,C;;MALJ,IA
AI,QzBq7JG,YAAQ,CyBr7Jf,C;QAAwB,OAAO,S;MAC/B,qD;IAMJ,C;;MAcW,wC;MAAA,4B;IAAA,C;;MAMi
C,qB;QAAE,OAAM,aAAN,mB;MAAF,C;IAAA,C;;MAJhC,YAAqB,4BAAT,qBAAS,C;MACrB,IAAI,KAAM,U
AAV,C;QACI,OAAO,eAAW,W;;QAEIB,OAAkB,YAAX,eAAW,EAAU,4CAAV,CAA0B,W;IACpD,C;;MAPJ,q
D;IASJ,C;;MAcW,wC;MAAA,4B;IAAA,C;;MAMiC,qB;QAAE,OAAM,aAAN,mB;MAAF,C;IAAA,C;;MAJhC,Y
AAqB,YAAT,qBAAS,C;MACrB,IAAI,KAAM,UAAV,C;QACI,OAAO,eAAW,W;;QAEIB,OAAkB,YAAX,eAAW
,EAAU,4CAAV,CAA0B,W;IACpD,C;;MAPJ,qD;IASJ,C;8FAEA,yB;MAAA,4C;MAAA,qC;QAOI,OAAO,iBAA
M,OAAAN,C;MACX,C;KARA,C;wFAUA,yB;MAAA,+D;MAAA,6B;MAAA,uC;QAUoB,Q;QAFhB,YAAy,gB;Q
ACZ,aAAa,gB;QACG,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,OAAV,CAAJ,C;YACI,KA
AM,WAAL,OAAJ,C;;YAEN,MAAO,WAAL,OAAJ,C;;QAGf,OAAO,cAAK,KAAL,EAAy,MAAZ,C;MACX,C;K

AIBA,C;;MA0BI,OAA6C,UAAtC,YAAW,SAAX,EAAiB,YAAW,OAAx,EAAjB,EAAc,C;IACjD,C;;MAWI,OA
AY,OAAAL,SAAK,EAAc,OAAAT,QAAS,CAAd,C;IACbB,C;;MAWI,OAA+C,UAAxC,YAAW,SAAX,EAA0B,aAA
T,QAAS,CAA1B,EAAwC,C;IACnD,C;;MAWI,OAAkC,UAA3B,YAAW,SAAX,EAAiB,QAAjB,EAA2B,C;IACtC
,C;4FAEA,yB;MAAA,0C;MAAA,qC;QAOI,OAAO,gBAAK,OAAAL,C;MACX,C;KARA,C;;MA0B+C,oB;QAAA,
OAAy,C;MAAG,8B;QAAA,iBAA0B,K;MACpF,OAAO,8BAAiB,IAAjB,EAAuB,IAAvB,EAA6B,cAA7B,EAA2
D,KAA3D,C;IACX,C;;MAObkD,oB;QAAA,OAAy,C;MAAG,8B;QAAA,iBAA0B,K;MACvF,OAAwE,OAAjE,8
BAAiB,IAAjB,EAAuB,IAAvB,EAA6B,cAA7B,EAA2D,IAA3D,CAAiE,EAAI,SAAJ,C;IAC5E,C;;MAWoD,cAA
M,EAAAN,C;IAAS,C;;MAAZD,OAAO,oBAAGB,SAAhB,EAAcB,KAAtB,EAA6B,UAA7B,C;IACX,C;;MAYI,OA
AO,oBAAGB,SAAhB,EAAcB,KAAtB,EAA6B,SA7B,C;IACX,C;;MAaiC,aAAK,CAAL,C;IAAO,C;;MAApC,O
AAO,yBAAY,kBAAZ,C;IACX,C;;MAc4B,wC;MAAA,6B;MAAA,yB;MAAA,gD;MAAA,kD;MAAA,4B;MAAA,
2B;MAAA,wB;MAAA,kC;IASxB,C;;oCARmB,sC;cACf,IAAI,CAAC,mBAAS,UAAAd,C;gBAAyB,M;;gBAA
zB,gB;;mCACc,mBAAS,O;cACvB,gB;;cAAA,KAAO,mBAAS,UAAhB,C;gBAAA,gB;;gCACe,mBAAS,O;c
ACpB,gB;8BAAA,iCAAM,6BAAU,kBAAV,EAAmB,eAAnB,CAAN,O;kBAAA,kBCpqDZ,yB;uBDqDY,yB;cA
AA,Q;cACA,qBAAU,e;cAHd,gB;;cAKJ,W;;IAAA,C;;MATwB,yD;uBAAA,4G;YAAA,S;iBAAA,Q;;iB
AAA,uB;MAAA,C;IAAA,C;;MAAxB,OAAO,SAAS,0CAAT,C;IAUX,C;;MAY6D,yB;QAAA,YAA0B,I;MAAM,s
B;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;MAAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAA
O,yB;QAAA,YAAoC,I;MAGtN,Q;MAFhB,MAAO,gBAAO,MAAP,C;MACP,YAAy,C;MACI,2B;MAAhB,OAAg
B,cAAhB,C;QAAgB,yB;QACZ,IAAI,iCAAU,CAAd,C;UAAiB,MAAO,gBAAO,SAAP,C;QACxB,IAAI,QAAQ,C
AAR,IAAa,SAAS,KAA1B,C;UACW,gBAAP,MAAO,EAAc,OAAAd,EAAuB,SAAvB,C;;UACJ,K;;MAEX,IAAI,S
AAS,CAAT,IAAc,QAAQ,KAA1B,C;QAAiC,MAAO,gBAAO,SAAP,C;MACxC,MAAO,gBAAO,OAAp,C;MACP
,OAAO,M;IACX,C;;MAYwC,yB;QAAA,YAA0B,I;MAAM,sB;QAAA,SAAuB,E;MAAI,uB;QAAA,UAAwB,E;M
AAI,qB;QAAA,QAAa,E;MAAI,yB;QAAA,YAA0B,K;MAAO,yB;QAAA,YAAoC,I;MACjN,OAAO,oBAAO,sBA
AP,EAAwB,SAAxB,EAAmC,MAAnC,EAA2C,OAA3C,EAAoD,KAApD,EAA2D,SA3D,EAAcE,SAAtE,CAAiF
,W;IAC5F,C;;MAMoB,mB;QAAE,OAAA,eAAK,W;MAAP,C;IAAA,C;;MAAhB,oCAAGB,8BAAhB,C;IACJ,C;;
MAOI,OAAO,S;IACX,C;;MAWoB,Q;MAFhB,UAAkB,G;MACIB,YAAiB,C;MACD,2B;MAAhB,OAAgB,cAAh
B,C;QAAgB,yB;QACZ,OAAO,O;QACP,oBAAmB,qBAAnB,EAAmB,KAAAnB,E;;MAEJ,OAAW,UAAc,CAAb,G
AAgB,wCAAO,IAAvB,GAAgC,MAAM,K;IACjD,C;;MAWoB,Q;MAFhB,UAAkB,G;MACIB,YAAiB,C;MACD,2
B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;QACP,oBAAmB,qBAAnB,EAAmB,KAAAnB,E;;MAE
J,OAAW,UAAc,CAAb,GAAgB,wCAAO,IAAvB,GAAgC,MAAM,K;IACjD,C;;MAWoB,Q;MAFhB,UAAkB,G;M
ACIB,YAAiB,C;MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;QACP,oBAAmB,qBAAnB,
EAAmB,KAAAnB,E;;MAEJ,OAAW,UAAc,CAAb,GAAgB,wCAAO,IAAvB,GAAgC,MAAM,K;IACjD,C;;MAWo
B,Q;MAFhB,UAAkB,G;MACIB,YAAiB,C;MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;
QACP,oBAAmB,qBAAnB,EAAmB,KAAAnB,E;;MAEJ,OAAW,UAAc,CAAb,GAAgB,wCAAO,IAAvB,GAAgC,
MAAM,K;IACjD,C;;MAWoB,Q;MAFhB,UAAkB,G;MACIB,YAAiB,C;MACD,2B;MAAhB,OAAgB,cAAhB,C;Q
AAgB,yB;QACZ,OAAO,O;QACP,oBAAmB,qBAAnB,EAAmB,KAAAnB,E;;MAEJ,OAAW,UAAc,CAAb,GAAgB
,wCAAO,IAAvB,GAAgC,MAAM,K;IACjD,C;;MAWoB,Q;MAFhB,UAAkB,G;MACIB,YAAiB,C;MACD,2B;MA
AhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,O;QACP,oBAAmB,qBAAnB,EAAmB,KAAAnB,E;;MAEJ,OA
AW,UAAc,CAAb,GAAgB,wCAAO,IAAvB,GAAgC,MAAM,K;IACjD,C;;MAUoB,Q;MADhB,UAAe,C;MACC,2
B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,YAAO,O;;MAEX,OAAO,G;IACX,C;;MAUoB,Q;MADhB,UA
Ae,C;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,YAAO,O;;MAEX,OAAO,G;IACX,C;;MAUoB,Q
;MADhB,UAAe,C;MACC,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,YAAO,OAAp,I;;MAEJ,OAAO,G;I
ACX,C;;MAUoB,Q;MADhB,Y;MACgB,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,cAAO,OAAp,C;;MA
EJ,OAAO,G;IACX,C;;MAUoB,Q;MADhB,UAAiB,G;MACD,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
OAAO,O;;MAEX,OAAO,G;IACX,C;;MAUoB,Q;MADhB,UAAkB,G;MACF,2B;MAAhB,OAAgB,cAAhB,C;QA
AgB,yB;QACZ,OAAO,O;;MAEX,OAAO,G;IACX,C;;MEz3DI,aAAa,qBAAiB,YAAy,cAAZ,CAAjB,C;MACb,kB
AAc,KAAAd,C;MR+sBgB,Q;MAAA,OQ9sBT,SR8sBS,W;MAAhB,OAAgB,cAAhB,C;QAAgB,2B;QAAU,oB;QQ9
sBK,IAAI,CAAC,SAAD,IAAY,OR8sBX,SQ9sBW,UAAhB,C;UAAiC,YAAU,I;UAA3C,mBAAiD,K;;UAAjD,m
BAA8D,I;;QR8sBvE,qB;UQ9sBD,MR8sBqC,WAAI,SAAJ,C;;MQ9sB1D,OAAqB,M;IACzB,C;;MAWI,aAAa,qB

AAiB,SAAjB,C;MACN,YAAP,MAAO,EAAU,QAAV,C;MACP,OAAO,M;IACX,C;;MAWI,YAAqB,gCAAT,QAAS,EAAGC,SAAhC,C;MACrB,IAAI,KAAM,UAAV,C;QACI,OAAy,QAAL,SAAK,C;MAChB,IAAI,yBAAJ,C;QACgB,kBAAY,sB;QRwqBZ,Q;QAAA,OQxqBL,SRwqBK,W;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CQxqBwB,qBRwqBb,OQxqBa,CRwqB5B,C;YAAyB,WAAy,WAAI,OAAJ,C;;QQxqBvD,ORYqBG,W;;MQxqBP,aAAa,qBAAiB,SAAjB,C;MACb,MAAO,mBAAU,KAAV,C;MACP,OAAO,M;IACX,C;;MAWI,aAAa,qBAAiB,SAAjB,C;MACN,YAAP,MAAO,EAAU,QAAV,C;MACP,OAAO,M;IACX,C;gGAEA,yB;MAAA,8C;MAAA,qC;QAOI,OAAO,iBAAM,OAAN,C;MACX,C;KARA,C;;MAGBI,aAAa,qBAAiB,YAAy,iBAAO,CAAP,IAAZ,CAAjB,C;MACb,MAAO,gBAAO,SAAP,C;MACP,MAAO,WAAI,OAAJ,C;MACP,OAAO,M;IACX,C;;MASI,aAAa,qBAAiB,YAAy,SAAK,KAAL,GAAY,QAAS,OAArB,IAAZ,CAAjB,C;MACb,MAAO,gBAAO,SAAP,C;MACA,SAAP,MAAO,EAAO,QAAP,C;MACP,OAAO,M;IACX,C;;MAQuD,UAAU,M;MAA1C,aAAa,qBAAiB,YAAy,WAAS,4BAAT,QAAS,CAAT,YAA4C,cAAL,WAAvC,4BAA2D,SAAK,KAAL,GAAY,CAAZ,IAAvE,CAAjB,C;MACb,MAAO,gBAAO,SAAP,C;MACA,OAAP,MAAO,EAAO,QAAP,C;MACP,OAAO,M;IACX,C;;MASI,aAAa,qBAAiB,YAAy,SAAK,KAAL,GAAY,CAAZ,IAAZ,CAAjB,C;MACb,MAAO,gBAAO,SAAP,C;MACA,SAAP,MAAO,EAAO,QAAP,C;MACP,OAAO,M;IACX,C;8FAEA,yB;MAAA,4C;MAAA,qC;QAOI,OAAO,gBAAK,OAAL,C;MACX,C;KARA,C;;MhB1HuF,wC;IANvF,C;IAAA,0D;MAO2C,8B;IAAA,C;IAP3C,gF;;MCSuF,wC;IARvF,C;IAAA,0D;MAS2C,8B;IAAA,C;IAT3C,gF;gGgBcA,yB;MAAA,uD;MAAA,gC;MAAA,iD;QAOI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAc,qBAAL,KAAL,CAAT,C,GAAcD,uBAAa,KAAb,E;MACjE,C;KARA,C;gGAUA,yB;MAAA,+C;MAAA,mC;QAOI,OAAy,UAAL,SAAK,EAAU,KAAV,C;MAChB,C;KARA,C;0EAUA,yB;MA0CA,6C;MAAA,oC;MAAA,gC;MA1CA,uC;QAKW,sB;;UAyCS,Q;UAAA,0B;UAAhB,OAAgB,cAAhB,C;YAAgB,oC;YAAM,IAzCH,SAyCO,CAAU,oBAAV,CAAJ,C;cAAwB,qBAAO,O;cAAP,uB;;UAC9C,qBAAO,I;;QA1CP,yB;MACJ,C;KANA,C;kFAQA,yB;MA8GA,mD;MAAA,+C;MAAA,oC;MA9GA,uC;QAKW,qB;;UA6GO,Q;UAAA,OAAa,SAAR,sBAAQ,CAAb,W;UAAAd,OAAC,cAAAd,C;YAAc,uB;YACV,cAAc,qBAAK,KAAL,C;YACd,IA/Gc,SA+GV,CAAU,oBAAV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;UAE5B,oBAAO,I;;QAJHP,wB;MACJ,C;KANA,C;;MAaI,ICwKgD,qBAAU,CDxK1D,C;QACI,MAAM,2BAAuB,yBAAvB,C;MACV,OAAO,qBAAK,CAAL,C;IACX,C;4EAEA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,iE;MAAA,uC;QAKoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,6DAAvB,C;MACV,C;KAPA,C;;MAaI,OCsJgD,qBAAU,CDtJnD,GAAe,IAAf,GAAY,qBAAK,CAAL,C;IACpC,C;wFAEA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,uC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,OAAO,O;;QACrD,OAAO,I;MACX,C;KANA,C;mFAQA,yB;MAAA,uD;MAAA,gC;MAAA,iD;QAKI,OAAW,SAAS,CAAT,IAAc,SAAS,wBAA3B,GAAc,qBAAL,KAAL,CAAT,C,GAAcD,uBAAa,KAAb,E;MACjE,C;KANA,C;;MAYI,OAAW,SAAS,CAAT,IAAc,SAAS,2BAA3B,GAAc,qBAAL,KAAL,CAAT,C,GAAcD,I;IACjE,C;0FAEA,yB;MAAA,mD;MAAA,oC;MAAA,uC;QAIkB,gC;QAAA,6B;QAAA,mB;QAAA,kB;QAAA,kB;QAAd,0D;UACI,IAAI,UAAU,iCAAK,KAAL,EAAV,CAAJ,C;YACI,OAAO,K;;QAGf,OAAO,E;MACX,C;KAVA,C;wFAYA,yB;MAAA,mD;MAAA,+C;MAAA,oC;MAAA,uC;QAIkB,Q;QAAA,OAAQ,SAAR,sBAAQ,CAAR,W;QAAd,OAAC,cAAAd,C;UAAc,uB;UACV,IAAI,UAAU,iCAAK,KAAL,EAAV,CAAJ,C;YACI,OAAO,K;;QAGf,OAAO,E;MACX,C;KAVA,C;;MAiBI,IC+FgD,qBAAU,CD/F1D,C;QACI,MAAM,2BAAuB,yBAAvB,C;MACV,OAAO,qBAAK,2BAAL,C;IACX,C;0EAEA,yB;MAAA,mD;MAAA,+C;MAAA,oC;MAAA,iE;MAAA,uC;QAKkB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,CAAQ,CAAb,W;QAAd,OAAC,cAAAd,C;UAAc,uB;UACV,cAAc,qBAAK,KAAL,C;UACd,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,OAAO,O;;QAEnC,MAAM,gCAAuB,6DAAvB,C;MACV,C;KAVA,C;;MAGBI,OC0EgD,qBAAU,CD1EnD,GAAe,IAAf,GAAY,qBAAK,mBAAS,CAAT,IAAL,C;IACpC,C;sFAEA,yB;MAAA,mD;MAAA,+C;MAAA,oC;MAAA,uC;QAIkB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,CAAQ,CAAb,W;QAAd,OAAC,cAAAd,C;UAAc,uB;UACV,cAAc,qBAAK,KAAL,C;UACd,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,OAAO,O;;QAEnC,OAAO,I;MACX,C;KATA,C;8EAWA,yB;MAAA,mC;MAAA,yC;MAAA,4B;QAQI,OAAO,kBAAO,cAAP,C;MACX,C;KATA,C;;MAKBI,IC0CgD,qBAAU,CD1C1D,C;QACI,MAAM,2BAAuB,yBAAvB,C;MACV,OAAO,qBAAL,MAAO,iBAAQ,gBAAR,CAAX,C;IACX,C;;MAMiB,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UAAK,MAAM,2BAAuB,yBAAvB,C;aACX,C;UAAK,4BAAK,CAAL,C;UAAL,K;gBACQ,MAAM,gCAAyB,0CAAzB,C;;MAHIB,W;IAKJ,C;8EAEA,yB;MAAA,6C;MAAA,oC;MAAA,kF;MAAA,gC;MAAA,iE;MAAA,8B;MAAA,uC;QAMoB,UAST,M;QAXP,aAAoB,I;QACpB,YA

AY,K;QACI,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,IAAI,UAAU,oBAAV,CAAJ,C;YACI,IAAI,KAJ
,C;cAAW,MAAM,8BAAYB,wDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAA
Y,MAAM,gCAAuB,6DAAvB,C;QAEIB,OAAO,4E;MACX,C;KAhBA,C;;MAsBI,OAAW,qBAAU,CAAd,GAAiB,
qBAAK,CAAL,CAAjB,GAA8B,I;IACzC,C;0FAEA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,uC;QAMoB,Q;
QAFhB,aAAoB,I;QACpB,YAAy,K;QACI,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,IAAI,UAAU,oBAA
V,CAAJ,C;YACI,IAAI,KAJ,C;cAAW,OAAO,I;YACIB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,
C;UAAy,OAAO,I;QACnB,OAAO,M;MACX,C;KafA,C;;MfrMI,IAAI,Ee8NI,KAak,Cf9NT,CAAJ,C;QACI,ce6N
c,wD;Qf5Nd,MAAM,gCAAyB,OAAQ,WAAjC,C;;Me6NV,OAAO,8BAAc,eAAF,CAAE,EAAa,gBAAb,CAAd,E
AAoC,gBAAP,C;IACX,C;;MfhOI,IAAI,Ee0OI,KAak,Cf1OT,CAAJ,C;QACI,ceyOc,wD;QfxOd,MAAM,gCAAy
B,OAAQ,WAAjC,C;;MeyOV,OE3E6E,oBF2E1D,eAAF,CAAE,EAAa,gBAAb,CE3E0D,C;IF4EjF,C;;Mf5OI,IAAI,
EesPI,KAak,CftPT,CAAJ,C;QACI,ceqPc,wD;QfpPd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MeqPV,OAAO,mBA
AkB,gBAAZ,mBAAS,CAAT,IAAY,EAac,CAAd,CAAlB,C;IACX,C;;MfxPI,IAAI,EekQI,KAak,CflQT,CAAJ,C;
QACI,ceiQc,wD;QfhQd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MeiQV,OAAO,mBAakB,gBAAZ,mBAAS,CAAT,I
AAy,EAac,CAAd,CAAlB,C;IACX,C;2FAEA,yB;MAAA,uD;MAAA,oC;MAAA,uC;QAMI,iBAAc,wBAAd,WA
A+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAV,CAAL,C;YACI,OAAO,8BAAY,CAAZ,EAae,
QAAQ,CAAR,IAAf,C;QACf,OAAO,E;MACX,C;KAVA,C;4FAYA,yB;MAAA,uD;MAAA,oC;MAAA,uC;QAMI,
iBAAc,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAV,CAAL,C;YACI,OEvhof,o
BFuHnE,CEvHmE,EFuHhE,QAAQ,CAAR,IEvHgE,C;;QFwH5F,OAAO,E;MACX,C;KAVA,C;oFAYA,yB;MAA
A,mD;MAAA,oC;MAAA,uC;QAMuB,UAAL,MAAK,EAAL,MAAK,EAAL,M;QAAK,mBAAL,SAAK,C;QAAL,
mB;QAAA,kB;QAAA,kB;QAAd,0D;UACI,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAV,CAAL,C;YACI,OAAO,8
BAAY,KAaz,EAAMB,gBAAnB,C;QACf,OAAO,E;MACX,C;KAVA,C;oFAYA,yB;MAAA,mD;MAAA,oC;MA
AA,uC;QAMuB,UAAL,MAAK,EAAL,MAAK,EAAL,M;QAAK,mBAAL,SAAK,C;QAAL,mB;QAAA,kB;QAAA
,kB;QAAd,0D;UACI,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAV,CAAL,C;YACI,OElJqE,oBFkjpD,KElJoD,C;;QF
mJ7E,OAAO,E;MACX,C;KAVA,C;8EAYA,yB;MAAA,yD;MAkEA,oC;MAIEA,uC;QAIW,kBAAS,oB;QAKEM,
Q;QAAA,uB;QAAtB,iBAAc,CAAd,wB;UACI,cAAc,qBAAl,KAJ,C;UACd,IApE6B,SAoEzB,CAAU,oBAAV,C
AAJ,C;YAAwB,WAAY,gBAAO,OAAP,C;;QApExC,OAsEO,W;MArEX,C;KALA,C;8EAOA,yB;MAAA,yD;MA
2DA,oC;MA3DA,uC;QAIW,kBAAS,oB;QA2DM,Q;QAAA,uB;QAAtB,iBAAc,CAAd,wB;UACI,cAAc,qBAAl,K
AAJ,C;UACd,IA7D6B,SA6DzB,CAAU,oBAAV,CAAJ,C;YAAwB,WAAY,gBAAO,OAAP,C;;QA7DxC,OA+DO,
WA/DqC,W;MACHd,C;KALA,C;4FAOA,yB;MAAA,yD;MAkBA,gC;MAkoBA,6C;MAAA,oC;MAppBA,uC;QA
MW,kBAAGB,oB;QAqpBV,gB;QADb,YAAy,C;QACC,0B;QAAb,OAAa,cAAb,C;UAAa,iC;UAAM,eAAO,cAAP
,EAAO,sBAAP,S;UAAA,cAAgB,iB;UAlOB/B,IAnBoC,SAmBhC,CAAU,OAAV,EAAiB,OAAjB,CAAJ,C;YAA2
C,2BAAO,kBAAP,C;;QAnB/C,OAqBO,W;MApBX,C;KAPA,C;4FASA,yB;MAAA,yD;MASA,gC;MAkoBA,6C;
MAAA,oC;MA3oBA,uC;QAMW,kBAAGB,oB;QA4oBV,gB;QADb,YAAy,C;QACC,0B;QAAb,OAAa,cAAb,C;U
AAa,iC;UAAM,eAAO,cAAP,EAAO,sBAAP,S;UAAA,cAAgB,iB;UAlOB/B,IAVOC,SAUhC,CAAU,OAAV,EAAi
B,OAAjB,CAAJ,C;YAA2C,2BAAO,kBAAP,C;;QAV/C,OAYO,WAZ4C,W;MACvD,C;KAPA,C;gGASA,yB;MA
AA,gC;MAkoBA,6C;MAAA,oC;MAlOB,oD;QAYoBiB,gB;QADb,YAAy,C;QACC,0B;QAAb,OAAa,cAAb,C;U
AAa,iC;UAAM,eAAO,cAAP,EAAO,sBAAP,S;UAAA,cAAgB,iB;UAlOB/B,IAAI,UAAU,OAAV,EAAiB,OAAjB,
CAAJ,C;YAA2C,2BAAO,kBAAP,C;;QAE/C,OAAO,W;MACX,C;KAVA,C;oFAYA,yB;MAAA,yD;MAcA,6C;M
AAA,oC;MAAA,gC;MAdA,uC;QAIW,kBAAY,oB;QAcH,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;U
AAM,IAAI,CAdU,SACt,CAAU,oBAAV,CAAL,C;YAAyB,WAAY,gBAAO,OAAP,C;;QAd3D,OAeO,W;MAdX,
C;KALA,C;oFAOA,yB;MAAA,yD;MAOA,6C;MAAA,oC;MAAA,gC;MAPA,uC;QAIW,kBAAY,oB;QAOH,Q;Q
AAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,IAAI,CAPU,SAOT,CAAU,oBAAV,CAAL,C;YAAyB,
WAAY,gBAAO,OAAP,C;;QAP3D,OAQO,WARwC,W;MACnD,C;KALA,C;wFAOA,yB;MAAA,6C;MAAA,oC;
MAAA,gC;MAAA,oD;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,IAAI,CAAC,UAAU,
oBAAV,CAAL,C;YAAyB,WAAY,gBAAO,OAAP,C;;QAC3D,OAAO,W;MACX,C;KANA,C;kFAQA,yB;MAAA,
oC;MAAA,oD;QAI0B,Q;QAAA,uB;QAAtB,iBAAc,CAAd,wB;UACI,cAAc,qBAAl,KAJ,C;UACd,IAAI,UAAU,
oBAAV,CAAJ,C;YAAwB,WAAY,gBAAO,OAAP,C;;QAExC,OAAO,W;MACX,C;KATA,C;;MAeI,IAAI,OAAQ,
UAAZ,C;QAAuB,OAAO,E;MAC9B,OAAO,yBAAY,OAAZ,C;IACX,C;;MAMI,IAAI,OAAQ,UAAZ,C;QAAuB,O

AAO,E;MAC9B,OAAO,uBAAU,OAAV,C;IACX,C;;MASc,Q;MAHV,WAAmB,wBAAR,OAAQ,EAAwB,EAAXB
,C;MACnB,IAAI,SAAQ,CAAZ,C;QAae,OAAO,E;MACtB,aAAa,mBAAC,IAAd,C;MACH,yB;MAAV,OAAU,cA
AV,C;QAAU,mB;QACN,MAAO,gBAAO,qBAAL,CAAJ,CAAP,C;;MAEX,OAAO,M;IACX,C;4EAEA,yB;MAAA
,8B;MAAA,uC;MAAA,qC;QAKY,Q;QAAR,OAA8B,MAAtB,2DAAsB,EAAM,OAAO,CAAe,W;MACjD,C;KAN
A,C;;MfhaI,IAAI,EegbI,KAAK,CfhtT,CAAJ,C;QACI,ce+ac,wD;Qf9ad,MAAM,gCAAYB,OAAQ,WAAjC,C;;Me+
aV,OAAO,8BAAY,CAAZ,EAAiB,eAAF,CAAe,EAAa,gBAAb,CAAJB,C;IACX,C;;MflbI,IAAI,Ee4bI,KAAK,Cf5
bT,CAAJ,C;QACI,ce2bc,wD;Qf1bd,MAAM,gCAAYB,OAAQ,WAAjC,C;;Me2bV,OE1R4F,oBF0R3E,CE1R2E,EF
0RtE,eAAF,CAAe,EAAa,gBAAb,CE1RsE,C;IF2RhG,C;;Mf9bI,IAAI,EewcI,KAAK,CfxcT,CAAJ,C;QACI,ceucc,
wD;Qftcd,MAAM,gCAAYB,OAAQ,WAAjC,C;;MeucV,aAAa,gB;MACb,OAAO,8BAAY,SAAW,eAAF,CAAe,E
AAa,MAAb,CAAX,IAAZ,EAA6C,MAA7C,C;IACX,C;;Mf3cI,IAAI,EeqdI,KAAK,CfrdT,CAAJ,C;QACI,ceodc,w
D;Qfndd,MAAM,gCAAYB,OAAQ,WAAjC,C;;MeodV,aAAa,gB;MACb,OEvT6E,oBFuT5D,SAAW,eAAF,CAAe,
EAAa,MAAb,CAAX,IEvT4D,C;IFwTjF,C;2FAEA,yB;MAAA,uD;MAAA,oC;MAAA,uC;QAMI,iBAAC,wBAAd,
WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,iCAAK,KAAL,EAAY,CAAL,C;YACI,OAAO,8BAAY,QAAQ,CA
AR,IAAZ,EAAuB,gBAAvB,C;;QAGf,OAAO,8BAAY,CAAZ,EAAe,gBAaf,C;MACX,C;KAZA,C;4FAcA,yB;M
AAA,uD;MAAA,oC;MAAA,uC;QAMI,iBAAC,wBAAd,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,iCAAK,K
AAL,EAAY,CAAL,C;YACI,OEhVqE,oBFgVpD,QAAQ,CAAR,IEhVoD,C;;QFmV7E,OAAO,S;MACX,C;KAZA
,C;oFAcA,yB;MAAA,oC;MAAA,uC;QAM0B,Q;QAAA,uB;QAAtB,iBAAC,CAAd,wB;UACI,IAAI,CAAC,UAAU
,iCAAI,KAJ,EAAY,CAAL,C;YACI,OAAO,8BAAY,CAAZ,EAAe,KAaf,C;;QAEf,OAAO,8BAAY,CAAZ,EAA
e,gBAaf,C;MACX,C;KAXA,C;oFAaA,yB;MAAA,oC;MAAA,uC;QAM0B,Q;QAAA,uB;QAAtB,iBAAC,CAAd,w
B;UACI,IAAI,CAAC,UAAU,iCAAI,KAJ,EAAY,CAAL,C;YACI,OEwWof,oBFwWnE,CExWmE,EFwWhE,KE
xWgE,C;;QF0W5F,OAAO,S;MACX,C;KAXA,C;;MAiBI,OAAO,qBAAC,SAAd,CAAoB,U;IAC/B,C;kFAEA,yB;
MAAA,8B;MAAA,6C;MAAA,4B;QAKY,Q;QAAR,OAA8B,SAAtB,2DAAsB,CAAW,W;MAC7C,C;KANA,C;oF
AQA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MAkEA,6C;MAAA,oC;MAAA,gC;MAIEA,uC;QASI,eAAmC,cAAp
B,YAAY,gBAAZ,CAAoB,EAAC,EAAd,C;QAC5B,kBAAY,mBAAoB,QAAPB,C;QA+DH,Q;QAAA,0B;QAaHb,
OAAgB,cAAhB,C;UAAgB,oC;UACZ,WAhE8C,SAGe/B,CAAU,oBAAV,C;UrBjCnB,wBAAL,IAAK,MAAT,EAA
gB,IAAK,OAARb,C;;QqB/BA,OakeO,W;MAjEX,C;KAXA,C;wFAaA,yB;MAAA,0D;MAAA,yD;MAAA,uE;MA
yBA,6C;MAAA,oC;MAAA,gC;MAZBA,yC;QASI,eAAmC,cAApB,YAAY,gBAAZ,CAAoB,EAAC,EAAd,C;QAC
5B,kBAAC,mBAAuB,QAaVb,C;QAuBL,Q;QAAA,0B;QAaHb,OAAgB,cAAhB,C;UAAgB,oC;UACZ,WAAy,aA
xBuC,WAwBnC,CAAY,oBAAZ,CAAJ,EAA0B,oBAA1B,C;;QAxBhB,OA0BO,W;MAzBX,C;KAXA,C;wFAaA,y
B;MAAA,0D;MAAA,yD;MAAA,uE;MA0BA,6C;MAAA,oC;MAAA,gC;MA1BA,yD;QAQI,eAAmC,cAApB,YA
AY,gBAAZ,CAAoB,EAAC,EAAd,C;QAC5B,kBAAC,mBAAoB,QAAPB,C;QAyBL,Q;QAAA,0B;QAaHb,OAAg
B,cAAhB,C;UAAgB,oC;UACZ,WAAy,aA1BoC,WA0BhC,CAAY,oBAAZ,CAAJ,EA1BiD,cA0BvB,CAAe,oBAA
f,CAA1B,C;;QA1BhB,OA4BO,W;MA3BX,C;KAVA,C;4FAYA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,sD
;QAQoB,Q;QAAA,0B;QAaHb,OAAgB,cAAhB,C;UAAgB,oC;UACZ,WAAy,aAAI,YAAY,oBAAZ,CAAJ,EAA0
B,oBAA1B,C;;QAEhB,OAAO,W;MACX,C;KAZA,C;4FAcA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,sE;Q
AQoB,Q;QAAA,0B;QAaHb,OAAgB,cAAhB,C;UAAgB,oC;UACZ,WAAy,aAAI,YAAY,oBAAZ,CAAJ,EAA0B,
eAAe,oBAAf,CAA1B,C;;QAEhB,OAAO,W;MACX,C;KAZA,C;wFAcA,yB;MAAA,6C;MAAA,oC;MAAA,gC;M
AAA,oD;QA0oB,Q;QAAA,0B;QAaHb,OAAgB,cAAhB,C;UAAgB,oC;UACZ,WAAe,UAAU,oBAAV,C;UrBjCn
B,wBAAL,IAAK,MAAT,EAAgB,IAAK,OAARb,C;;QqBmCA,OAAO,W;MACX,C;KAXA,C;2FAaA,yB;MAAA,0
D;MAAA,yD;MAAA,uE;MAgBA,6C;MAAA,oC;MAAA,gC;MAhBA,2C;QAYI,aAAa,mBAA2C,cAApB,YAAY,
gBAAZ,CAAoB,EAAC,EAAd,CAA3C,C;QAYG,Q;QAAA,0B;QAaHb,OAAgB,cAAhB,C;UAAgB,oC;UAXO,M
AYP,aAAI,oBAAJ,EAZe,aAYF,CAAc,oBAAd,CAAb,C;;QAZhB,OAAuB,M;MAC3B,C;KAdA,C;8FAGBA,yB;M
AAA,6C;MAAA,oC;MAAA,gC;MAAA,wD;QAQoB,Q;QAAA,0B;QAaHb,OAAgB,cAAhB,C;UAAgB,oC;UACZ
,WAAy,aAAI,oBAAJ,EAAa,cAAc,oBAAd,CAAb,C;;QAEhB,OAAO,W;MACX,C;KAZA,C;;MAkBiB,Q;MAAA,
4B;MAAb,OAAa,cAAb,C;QAAa,iC;QACT,WAAy,WAAI,iBAAJ,C;;MAEhB,OAAO,W;IACX,C;;MAMI,OAAO,
2BAAa,eAAc,YAAY,gBAAZ,CAAd,CAAb,C;IACX,C;;MAMIb,IAAN,I;MAAA,QAAM,gBAAN,C;aACH,C;UA
AK,kB;UAAL,K;aACA,C;UAAK,cAAo,iCAAK,CAAL,EAAP,C;UAAL,K;gBACa,wBAAL,SAAK,C;UAHV,K;;
MAAP,W;IAKJ,C;;MAMI,OAAO,2BAAa,iBAAgB,gBAAhB,CAAb,C;IACX,C;;MAQiB,IAAN,I;MAAA,QAAM,

gBAAN,C;aACH,C;UAAK,iB;UAAL,K;aACA,C;UAAK,aAAM,iCAAK,CAAL,EAAN,C;UAAL,K;gBACQ,kCA
Aa,qBAAoB,YAAY,gBAAZ,CAApB,CAAb,C;UAHL,K;;MAAP,W;IAKJ,C;gFAEA,yB;MAAA,+D;MAOA,6C;M
AAA,oC;MAAA,gD;MAAA,gC;MAPA,uC;QAIW,kBAAU,gB;QAOD,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;
UAAgB,oC;UACZ,WAR6B,SAQIB,CAAU,oBAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;QAThB,OAWO,W
;MAVX,C;KALA,C;oFAOA,yB;MAAA,6C;MAAA,oC;MAAA,gD;MAAA,gC;MAAA,oD;QAIoB,Q;QAAA,0B;Q
AAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,WAAW,UAAU,oBAAV,C;UACC,OAAZ,WAAY,EAAO,IAAP,C;;
QAEhB,OAAO,W;MACX,C;KATA,C;gFAWA,yB;MAAA,wE;MAyBA,6C;MAAA,oC;MAAA,+D;MAAA,gC;M
AzBA,yC;QASW,kBAAU,oB;QAYBD,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,UA1BoD,W
A0B1C,CAAY,oBAAZ,C;UrBneP,U;UADP,YqBqee,WrBreH,WqBqewB,GrBrexB,C;UACL,IAAI,aAAJ,C;YACH
,aqBmeuC,gB;YAA5B,WrBleX,aqBkegC,GrBlehC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UqB+dA,iB;UACA,IAA
K,WAAI,oBAAJ,C;;QA5BT,OA8BO,W;MA7BX,C;KAVA,C;gFAYA,yB;MAAA,wE;MA8BA,6C;MAAA,oC;M
AAA,+D;MAAA,gC;MA9BA,yD;QAUW,kBAAU,oB;QA8BD,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,
oC;UACZ,UA/BiD,WA+BvC,CAAY,oBAAZ,C;UrBrfP,U;UADP,YqBufe,WrBvfH,WqBufwB,GrBvfxB,C;UACL,
IAAI,aAAJ,C;YACH,aqBqfuC,gB;YAA5B,WrBpfX,aqBofgC,GrBpfc,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UqB
ifA,iB;UACA,IAAK,WAjCyD,cAiCrD,CAAE,oBAAf,CAAJ,C;;QAjCT,OAmCO,W;MAICX,C;KAXA,C;oFAaA,y
B;MAAA,6C;MAAA,oC;MAAA,+D;MAAA,gC;MAAA,sD;QASoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UA
AgB,oC;UACZ,UAAU,YAAY,oBAAZ,C;UrBneP,U;UADP,YqBqee,WrBreH,WqBqewB,GrBrexB,C;UACL,IAAI
,aAAJ,C;YACH,aqBmeuC,gB;YAA5B,WrBleX,aqBkegC,GrBlehC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UqB+dA
,iB;UACA,IAAK,WAAI,oBAAJ,C;;QAET,OAAO,W;MACX,C;KafA,C;oFAiBA,yB;MAAA,6C;MAAA,oC;MA
AA,+D;MAAA,gC;MAAA,sE;QAUoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,UAAU,YAA
Y,oBAAZ,C;UrBrfP,U;UADP,YqBufe,WrBvfH,WqBufwB,GrBvfxB,C;UACL,IAAI,aAAJ,C;YACH,aqBqfuC,gB;
YAA5B,WrBpfX,aqBofgC,GrBpfc,EAAS,MAAT,C;YACA,e;;;YAEA,c;;UqBifA,iB;UACA,IAAK,WAAI,eAAe,
oBAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAhBA,C;qFAkBA,yB;MAAA,6C;MAAA,oC;MAAA,kC;MAAA,4
C;;QAQW,sC;QAAA,8C;MARX,C;MAAA,gE;QASwD,OAAgB,SAAhB,oBAAgB,C;MAAhB,C;MATxD,oE;QA
U+C,gCAAY,oBAAZ,C;MAAA,C;MAV/C,gF;MAAA,yC;QAQI,2D;MAIJ,C;KAZA,C;wEAca,yB;MAAA,gE;M
AuEA,6C;MAAA,oC;MAAA,gC;MAvEA,uC;QAOW,kBAAM,eAAa,gBAAb,C;QAqEA,Q;QAAA,0B;QAAb,OA
Aa,cAAb,C;UAAa,iC;UACT,WAAY,WAtEmB,SAsEf,CAAU,iBAAV,CAAJ,C;;QAtEhB,OAuEO,W;MAtEX,C;K
ARA,C;sFAUA,yB;MAAA,gE;MA+BA,6C;MAAA,oC;MAAA,gC;MA/BA,uC;QAOW,kBAaA,eAAa,gBAAb,C;
QAgCP,gB;QADb,YAAY,C;QACC,0B;QAAb,OAAa,cAAb,C;UAAa,iC;UACT,WAAY,WAjC0B,SAiCtB,EAAU,
cAAV,EAAU,sBAAV,WAAmB,iBAAnB,CAAJ,C;;QAjChB,OakCO,W;MAjCX,C;KARA,C;mGAUA,yB;MAAA
,+D;MAUA,gC;MAkKA,6C;MAAA,oC;MA5KA,uC;QAOW,kBAAoB,gB;QA4Kd,gB;QADb,YAAY,C;QACC,0B
;QAAb,OAAa,cAAb,C;UAAa,iC;UAIKsB,U;UAAA,cAVQ,SAUR,EakKT,cAIKS,EakKT,sBAIKS,WakKA,iBAI
KA,W;YAA6C,6B;;;QAVhF,OAWO,W;MAVX,C;KARA,C;uGAUA,yB;MAAA,gC;MAkKA,6C;MAAA,oC;MAI
KA,oD;QAYKiB,gB;QADb,YAAY,C;QACC,0B;QAAb,OAAa,cAAb,C;UAAa,iC;UAIKsB,U;UAAA,yBAkKT,cAI
KS,EakKT,sBAIKS,WakKA,iBAIKA,W;YAA6C,6B;;;QACHF,OAAO,W;MACX,C;KATA,C;oFAWA,yB;MAA
A,6C;MAAA,oC;MAAA,gC;MAAA,oD;QAQiB,UACiB,M;QAF9B,YAAY,C;QACC,0B;QAAb,OAAa,cAAb,C;U
AAa,iC;UACT,WAAY,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,iBAAnB,CAAJ,C;;QACHB,OAAO,W;MA
CX,C;KAXA,C;qFAaA,yB;MAAA,+D;MAQA,gC;MA2HA,6C;MAAA,oC;MAnIA,uC;QAKW,kBAAa,gB;QAKI
,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UA1HK,U;UAAA,cARe,SAQf,CA0HQ,oBA1HR,W;YAA
sC,6B;;;QAR3D,OASO,W;MARX,C;KANA,C;yFAQA,yB;MAAA,gC;MA2HA,6C;MAAA,oC;MA3HA,oD;QA+H
oB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UA1HK,U;UAAA,wBA0HQ,oBA1HR,W;YAA
sC,6B;;;QAC3D,OAAO,W;MACX,C;KAPA,C;4EASA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oD;QAKiB,Q;QAA
A,0B;QAAb,OAAa,cAAb,C;UAAa,iC;UACT,WAAY,WAAI,UAAU,iBAAV,CAAJ,C;;QACHB,OAAO,W;MACX,
C;KARA,C;;MAe4B,mB;QAAE,iC;MAAF,C;IAAA,C;;MAAxB,OAAO,qBAAiB,6BAAjB,C;IACX,C;wEAEA,yB
;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,uC;QAMoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UA
AM,IAAI,CAAC,UAAU,oBAAV,CAAL,C;YAAyB,OAAO,K;;QACTd,OAAO,I;MACX,C;KARA,C;;MAGBI,OA
AO,EC3rByC,qBAAU,CD2rBnD,C;IACX,C;wEAEA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,uC;QAMoB,
Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,OAAO,I;;Q

ACrD,OAAO,K;MACX,C;KARA,C;;MAeI,OAAO,gB;IACX,C;4EAEA,yB;MAAA,6C;MAAA,oC;MAAA,gC;M
AAA,uC;QAKoB,Q;QADhB,YAAy,C;QACI,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,IAAI,UAAU,oB
AAV,CAAJ,C;YAAwB,qB;;QAC9C,OAAO,K;MACX,C;KAPA,C;0EASA,yB;MAAA,6C;MAAA,oC;MAAA,gC;
MAAA,gD;QAKoB,Q;QADhB,kBAaKB,O;QACF,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,cAAc,UA
AU,WAAV,EAAuB,oBAAvB,C;;QACpC,OAAO,W;MACX,C;KAPA,C;wFASA,yB;MAAA,6C;MAAA,oC;MAA
A,gC;MAAA,gD;QASoB,UAA8B,M;QAF9C,YAAy,C;QACZ,kBAaKB,O;QACF,0B;QAAhB,OAAgB,cAAhB,C;
UAAgB,oC;UAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAhB,EAAgC,oBAAhC,C;;QACpC,OAAO
,W;MACX,C;KAXA,C;mFAaA,yB;MAAA,uD;MAAA,oC;MAAA,gD;QAOoC,Q;QAHhC,YAAy,wB;QACZ,kB
AAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,kCAAI,YAAJ,EAAI,oBAAJ,SAAV,EAAwB,WA
AxB,C;;QAEIB,OAAO,W;MACX,C;KAVA,C;IGAYA,yB;MAAA,uD;MAAA,oC;MAAA,gD;QAOI,YAAy,wB;Q
ACZ,kBAaKB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAaV,EAAiB,iCAAI,KAJ,EAAjB,EA
A6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAdA,C;gFagBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;
MAAA,oC;QAIoB,Q;QAAA,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UAAM,OAAO,oBAAP,C;;MAC1B,C;K
ALA,C;8FAOA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAOiB,UAAa,M;QAD1B,YAAy,C;QACC,0B;
QAAb,OAAa,cAAb,C;UAAa,iC;UAAM,QAAO,cAAP,EAAO,sBAAP,WAAgB,iBAAhB,C;;MACvB,C;KARA,C;;
MAgBiB,Q;MAFb,ICHyBgD,qBAAU,CDgyB1D,C;QAAe,OAAO,I;MACtB,UAAU,qBAaK,CAAL,C;MACG,kC;
MAAb,aAAU,CAAV,iB;QACI,QAAQ,qBAaK,CAAL,C;QACR,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEv
B,OAAO,G;IACX,C;4EAEA,yB;MAAA,uD;MAAA,oC;MAAA,sC;QAMI,IC/yBgD,qBAAU,CD+yB1D,C;UAAe,
OAAO,I;QACtB,cAAc,qBAaK,CAAL,C;QACd,gBAaQb,cAAL,SAaK,C;QACrB,IAAI,cAAa,CAAJB,C;UAAoB
,OAAO,O;QAC3B,eAAe,SAAS,oBAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,qBAaK,CAAL,C;
UACR,QAAQ,SAAS,cAAT,C;UACR,IAAI,2BAAW,CAAX,KAJ,C;YACI,UAAU,C;YACV,WAAW,C;;QAGn
B,OAAO,O;MACX,C;KApBA,C;;MA4BiB,Q;MAFb,ICn0BgD,qBAAU,CDm0B1D,C;QAAe,OAAO,I;MACtB,U
AAU,qBAaK,CAAL,C;MACG,kC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,qBAaK,CAAL,C;QACR,IAAI,UAAW
,SAAQ,gBAAR,EAAa,cAAb,CAAX,GAA6B,CAAJC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQiB,
Q;MAFb,IC1BgD,qBAAU,CDg1B1D,C;QAAe,OAAO,I;MACtB,UAAU,qBAaK,CAAL,C;MACG,kC;MAAb,aA
AU,CAAV,iB;QACI,QAAQ,qBAaK,CAAL,C;QACR,IAAI,MAAM,CAAV,C;UAAa,MAAM,C;;MAEvB,OAAO,
G;IACX,C;4EAEA,yB;MAAA,uD;MAAA,oC;MAAA,sC;QAMI,IC/1BgD,qBAAU,CD+1B1D,C;UAAe,OAAO,I;
QACtB,cAAc,qBAaK,CAAL,C;QACd,gBAaQb,cAAL,SAaK,C;QACrB,IAAI,cAAa,CAAJB,C;UAAoB,OAAO,
O;QAC3B,eAAe,SAAS,oBAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,qBAaK,CAAL,C;UACR,
QAAQ,SAAS,cAAT,C;UACR,IAAI,2BAAW,CAAX,KAJ,C;YACI,UAAU,C;YACV,WAAW,C;;QAGnB,OAA
O,O;MACX,C;KApBA,C;;MA4BiB,Q;MAFb,ICn3BgD,qBAAU,CDm3B1D,C;QAAe,OAAO,I;MACtB,UAAU,qB
AAK,CAAL,C;MACG,kC;MAAb,aAAU,CAAV,iB;QACI,QAAQ,qBAaK,CAAL,C;QACR,IAAI,UAAW,SAAQ,
gBAAR,EAAa,cAAb,CAAX,GAA6B,CAAJC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAQI,OC14BgD
,qBAAU,C;IDm4B9D,C;0EAEA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,uC;QAMoB,Q;QAAA,0B;QAAhB
,OAAgB,cAAhB,C;UAAgB,oC;UAAM,IAAI,UAAU,oBAAV,CAAJ,C;YAAwB,OAAO,K;;QACrD,OAAO,I;MA
CX,C;KARA,C;6EUA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,oC;QAKmC,Q;QAAA,0B;QAAhB,OAAg
B,cAAhB,C;UAAgB,oC;UAAM,OAAO,oBAAP,C;;QAArC,gB;MACJ,C;KANA,C;8EAQA,yB;MAAA,4F;MAAA
,uD;MAAA,oC;MAAA,gC;MAAA,uC;QAOqB,Q;QAHjB,IC35BgD,qBAAU,CD25B1D,C;UACI,MAAM,mCAA8
B,uCAA9B,C;QACV,kBAaKB,qBAaK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,wB
AAV,EAAuB,iCAAK,KAAL,EAAvB,E;;QAEIB,OAAO,W;MACX,C;KAXA,C;4FAaA,yB;MAAA,4F;MAAA,uD
;MAAA,oC;MAAA,gC;MAAA,uC;QAUqB,Q;QAHjB,IC36BgD,qBAAU,CD26B1D,C;UACI,MAAM,mCAA8B,u
CAA9B,C;QACV,kBAaKB,qBAaK,CAAL,C;QACD,+B;QAAjB,iBAAc,CAAd,yB;UACI,cAAc,oBAAU,KAaV,
EAAiB,wBAAjB,EAA8B,iCAAK,KAAL,EAA9B,E;;QAEIB,OAAO,W;MACX,C;KAdA,C;uFAgBA,yB;MAAA,u
D;MAAA,4F;MAAA,oC;MAAA,gC;MAAA,uC;QAM0B,UAEU,M;QAJhC,YAAy,wB;QACZ,IAAI,QAAQ,CAA
Z,C;UAAe,MAAM,mCAA8B,uCAA9B,C;QACrB,kBAaKB,sBAAI,YAAJ,EAAI,oBAAJ,Q;QACIB,OAAO,SAAS
,CAAhB,C;UACI,cAAc,oBAAU,kCAAI,cAAJ,EAAI,sBAAJ,WAAV,EAAwB,wBAAxB,E;;QAEIB,OAAO,W;MA
CX,C;KAXA,C;qGAaA,yB;MAAA,uD;MAAA,4F;MAAA,oC;MAAA,gC;MAAA,uC;QAS0B,Q;QAFtB,YAAy,w
B;QACZ,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,uCAA9B,C;QACrB,kBAaKB,sBAAI,YAAJ,EAAI,oBA

AJ,Q;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,oBAAU,KAaV,EAAiB,iCAAI,KAaJ,EAAjB,EAA6B,wBAA
7B,E;UACd,qB;;QAEJ,OAAO,W;MACX,C;KafA,C;4EAiBA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,sC;Q
AKoB,Q;QADhB,UAAe,C;QACC,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,YAAO,SAAS,oBAAT,CAA
P,I;;QAEJ,OAAO,G;MACX,C;KATA,C;wFAWA,yB;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,sC;QAKoB,Q;Q
ADhB,UAAkB,G;QACF,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,OAAO,SAAS,oBAAT,C;;QAEX,OA
AO,G;MACX,C;KATA,C;;MAAsBI,OAAO,sBAAS,IAAT,EAAe,IAAf,EAAsC,IAAtC,C;IACX,C;;MAkBI,OAAO,s
BAAS,IAAT,EAAe,IAAf,EAAsC,IAAtC,EAAwD,SAAxD,C;IACX,C;;MAAmC,OAAA,EAAG,W;IAAW,C;;MAA
7C,OAAO,6BAAgB,IAAhB,EAAsB,sBAAtB,C;IACX,C;;MAkBI,OAAO,8BAAiB,IAAjB,EAAuB,IAAvB,EAA8
C,IAA9C,EAAgE,SAAhE,C;IACX,C;oFAEA,yB;MAAA,yD;MAAA,6C;MAAA,oC;MAAA,gC;MAAA,6B;MAA
A,uC;QAQoB,Q;QAFhB,YAAY,oB;QACZ,aAAa,oB;QACG,0B;QAAhB,OAAgB,cAAhB,C;UAAgB,oC;UACZ,I
AAI,UAAU,oBAAV,CAAJ,C;YACI,KAAM,gBAAO,OAAP,C;;YAEN,MAAO,gBAAO,OAAP,C;;QAGf,OAAO,
cAAK,KAAL,EAAy,MAAZ,C;MACX,C;KAhBA,C;oFAkBA,yB;MAAA,yD;MAAA,6C;MAAA,oC;MAAA,gC;
MAAA,6B;MAAA,uC;QAQoB,Q;QAFhB,YAAY,oB;QACZ,aAAa,oB;QACG,0B;QAAhB,OAAgB,cAAhB,C;UA
AgB,oC;UACZ,IAAI,UAAU,oBAAV,CAAJ,C;YACI,KAAM,gBAAO,OAAP,C;;YAEN,MAAO,gBAAO,OAAP,
C;;QAGf,OAAO,cAAK,KAAM,WAAx,EAAuB,MAAO,WAA9B,C;MACX,C;KAhBA,C;;MAmCkD,OAAA,EA
AG,W;IAAW,C;;MADpB,oB;QAAA,OAAY,C;MAAG,8B;QAAA,iBAA0B,K;MACjF,OAAO,sBAAS,IAAT,EAA
e,IAAf,EAAqB,cAArB,EAAqC,eAArC,C;IACX,C;;MAoBgD,oB;QAAA,OAAY,C;MAAG,8B;QAAA,iBAA0B,K;
MAOhE,Q;MANrB,oBAAoB,IAApB,EAA0B,IAA1B,C;MACA,eAAe,SAAK,O;MACpB,aAAa,iBAAa,CAAC,W
AAW,IAAX,GAaKb,CAAIb,IAAD,IAAwB,IAAxB,IAAb,C;MACb,YAAY,C;MACZ,OAAO,QAAQ,QAAf,C;QA
CI,UAAU,QAAQ,IAAR,I;QACO,IAAI,MAAM,QAAV,C;UAAsB,IAAI,cAAJ,C;YAAoB,e;;YAAc,K;;UAAa,U;Q
AAf,qB;QACA,MAAO,WAAI,UAAU,8BAAy,KAAZ,EAAMb,UAAAnB,CAAV,CAAJ,C;QACP,gBAAS,IAAT,I;
;MAEJ,OAAO,M;IACX,C;;MAmB0D,OAAA,EAAG,W;IAAW,C;;MADpB,oB;QAAA,OAAY,C;MAAG,8B;QA
AA,iBAA0B,K;MACzF,OAAO,8BAAiB,IAAjB,EAAuB,IAAvB,EAA6B,cAA7B,EAA6C,uBAA7C,C;IACX,C;;M
AuBoC,wB;QAAW,yBAAU,0CAAy,KAAZ,EAaKc,eAAAd,QAAQ,YAAR,IAAc,EAAa,4BAAb,CAAIC,CAAV,C;
MAAX,C;IAAA,C;;MAHoB,sB;QAAA,SAAY,C;MAAG,8B;QAAA,iBAA0B,K;MAC7F,oBAAoB,IAApB,EAA0
B,MAA1B,C;MACA,cAAc,KAaK,cAAJ,GAAoB,yBAApB,GAAiC,WAAQ,mBAAS,IAAT,GAAGb,CAAhB,IAA
R,CAAIC,EAaKe,MAAIE,C;MACd,OAA4B,OAAb,aAAR,OAQ,CAAa,EAAI,qDAAJ,C;IAChC,C;;MAoBI,aVh
vCO,MAAK,KUgvCY,gBVhvCZ,EUquCD,KAW2B,OVhvC1B,C;MUivCZ,WAAW,iBAAa,MAAb,C;MACX,aA
AU,CAAV,MAAkB,MAAIB,M;QACI,IAAK,WAdqB,GAcP,iCAAK,CAAL,EAoD,EAcE,YAdrB,KAcqB,YAAM,
CAAN,EAfD,CaCrB,C;;MAdT,OAgoB,I;IAfX,C;wEAEA,yB;MAAA,gE;MAAA,oC;MV/uCA,iB;MU+uCA,8C;Q
AQI,aVhvCO,MAAK,KUgvCO,SAAK,OVhvCZ,EUgvCoB,KAAM,OVhvC1B,C;QUivCZ,WAAW,eAAa,MAAb,
C;QACX,aAAU,CAAV,MAAkB,MAAIB,M;UACI,IAAK,WAAI,UAAU,iCAAK,CAAL,EAIV,EAAMb,6BAAM
,CAAN,EAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAdA,C;;MAyBW,sB;;QAaP,WAAW,mBAAS,CAAT,I;QA
CX,IAAI,OAAO,CAAX,C;UAAc,qBAAO,W;UAAp,uB;;QACd,aAAa,iBAAa,IAAb,C;QACb,iBAAc,CAAd,UAA
sB,IAAtB,U;UACI,MAAO,WAjBkB,GaiBJ,iCAAK,KAAL,EAjBI,EAiBS,iCAAK,QAAQ,CAAR,IAAL,EAjBT,C
AiBiB,C;;QAEX,qBAAO,M;;MANBP,yB;IACJ,C;uFAEA,yB;MAAA,qD;MAAA,gE;MAAA,oC;MAAA,uC;QAU
I,WAAW,mBAAS,CAAT,I;QACX,IAAI,OAAO,CAAX,C;UAAc,OAAO,W;QACrB,aAAa,eAAa,IAAb,C;QACb,i
BAAc,CAAd,UAAsB,IAAtB,U;UACI,MAAO,WAAI,UAAU,iCAAK,KAAL,EAIV,EAAMb,iCAAK,QAAQ,CAA
R,IAAL,EAAvB,CAAJ,C;;QAEX,OAAO,M;MACX,C;KAjBA,C;;MAwBoB,mB;QAAE,OAaK,WAAI,eAAK,C;
MAAP,C;IAAA,C;;MADZ,wC;MAAA,S;QAAkB,OC3uC0B,qBAAU,C;;MD2uC1D,S;QAAiC,OAAO,W;MACxC
,oCAAgB,8BAAhB,C;IACJ,C;;MAOoB,mB;QAAE,OAaK,WAAI,eAAK,C;MAAP,C;IAAA,C;;MADZ,wC;MAA
A,S;QAAkB,OCnvC0B,qBAAU,C;;MDmvC1D,S;QAAiC,OAAO,e;MACxC,oCAAgB,8BAAhB,C;IACJ,C;;MGn5
CsC,wB;QAAW,OAAA,aAAK,KAAL,C7B8FV,K;M6B9FD,C;IAAA,C;;MCAF,wB;QAAW,OAAA,aAAK,KAAL
,C7B0HV,K;M6B1HD,C;IAAA,C;;MCAE,wB;QAAW,OAAA,aAAK,KAAL,CCqIV,K;MDrID,C;IAAA,C;;MEAE
,wB;QAAW,OAAA,aAAK,KAAL,C/ByGV,K;M+BzGD,C;IAAA,C;;MChDpC,OAAO,sBAAL,CAAJ,C;IACX,C;;
MASI,OAAO,sBAAL,CAAJ,C;IACX,C;;MASI,OAAO,sBAAL,CAAJ,C;IACX,C;;MASI,OAAO,sBAAL,CAAJ,C;I
ACX,C;;MASI,OAAO,sBAAL,CAAJ,C;IACX,C;;MASI,OAAO,sBAAL,CAAJ,C;IACX,C;;MASI,OAAO,s
BAAL,CAAJ,C;IACX,C;;MASI,OAAO,sBAAL,CAAJ,C;IACX,C;;MASI,OAAO,s

BAAI,CAAJ,C;IACX,C;;MASI,OAAO,sBAAI,CAAJ,C;IACX,C;;MASI,OAAO,sBAAI,CAAJ,C;IACX,C;;MASI,
OAAO,sBAAI,CAAJ,C;IACX,C;;MASI,OAAO,sBAAI,CAAJ,C;IACX,C;;MASI,OAAO,sBAAI,CAAJ,C;IACX,C;
;MASI,OAAO,sBAAI,CAAJ,C;IACX,C;;MASI,OAAO,sBAAI,CAAJ,C;IACX,C;;MASI,OAAO,sBAAI,CAAJ,C;I
ACX,C;;MASI,OAAO,sBAAI,CAAJ,C;IACX,C;;MASI,OAAO,sBAAI,CAAJ,C;IACX,C;uGAsCA,yB;MAkgGI,8
D;MAIlgGJ,iD;QASe,oBAAS,C;QAAT,S;UAAc,gBAy/FT,cAAR,iBAAQ,C;;QAz/FhB,OAAO,OAAcS,sBAAI,KA
AJ,CAAtC,GAAsD,aAAa,KAAb,C;MACjE,C;KAVA,C;uGAYA,yB;MA8/Fl,8D;MA9/FJ,iD;QASe,oBAAS,C;QA
AT,S;UAAc,gBAq/FT,cAAR,iBAAQ,C;;QAr/FhB,OAAO,OAAcS,sBAAI,KAJ,CAAtC,GAAsD,aAAa,KAAb,C;
MACjE,C;KAVA,C;uGAYA,yB;MA0/Fl,8D;MA1/FJ,iD;QASe,oBAAS,C;QAAT,S;UAAc,gBAi/FT,cAAR,iBAA
Q,C;;Qaj/FhB,OAAO,OAAcS,sBAAI,KAJ,CAAtC,GAAsD,aAAa,KAAb,C;MACjE,C;KAVA,C;uGAYA,yB;M
As/Fl,8D;MAt/FJ,iD;QASe,oBAAS,C;QAAT,S;UAAc,gBA6+FT,cAAR,iBAAQ,C;;QA7+FhB,OAAO,OAAcS,sB
AAI,KAJ,CAAtC,GAAsD,aAAa,KAAb,C;MACjE,C;KAVA,C;uGAYA,yB;MAAA,sD;MAAA,mC;QASI,OAA
Y,UAAL,SAAK,EAAU,KAAV,C;MACHb,C;KAVA,C;uGAYA,yB;MAAA,sD;MAAA,mC;QASI,OAA Y,UAAL,
SAAK,EAAU,KAAV,C;MACHb,C;KAVA,C;uGAYA,yB;MAAA,sD;MAAA,mC;QASI,OAA Y,UAAL,SAAK,EA
AU,KAAV,C;MACHb,C;KAVA,C;uGAYA,yB;MAAA,sD;MAAA,mC;QASI,OAA Y,UAAL,SAAK,EA
AU,KAAV,C;MACHb,C;KAVA,C;uGAYA,yB;MAAA,sD;MAAA,mC;QASI,OAA Y,UAAL,SAAK,EAAU,KAA
V,C;MACHb,C;KAVA,C;;MAmBW,sB;;QAgNS,Q;QAAA,2B;QAAhB,OAAGb,cAAhB,C;UAAgB,yB;UAAM,IA
hNH,SAgNO,CAAU,OA AV,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;MAjNP,yB;IACJ,C;;M
ASW,sB;;QAiNS,Q;QAAA,2B;QAAhB,OAAGb,cAAhB,C;UAAgB,yB;UAAM,IAjNH,SAiNO,CAAU,OA AV,CA
AJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;MALNP,yB;IACJ,C;;MASW,sB;;QAKNS,Q;QAAA,2B;
QAAhB,OAAGb,cAAhB,C;UAAgB,yB;UAAM,IAINH,SAkNO,CAAU,OA AV,CAAJ,C;YAAwB,qBAAO,O;YAA
P,uB;;;QAC9C,qBAAO,I;;MAnNP,yB;IACJ,C;;MASW,sB;;QAmNS,Q;QAAA,2B;QAAhB,OAAGb,cAAhB,C;U
AAgB,yB;UAAM,IA nNH,SA mNO,CAAU,OA AV,CAAJ,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;
MApNP,yB;IACJ,C;yFAEA,yB;MA4kBA,+C;MA8wEI,0D;MA11FJ,uC;QAOW,qB;;UA4kBO,Q;UAAA,OAAa,S
AuwEX,YAAR,iBAAQ,CAvwEW,CAAb,W;UAAAd,OAAC,cAAAd,C;YAAc,uB;YACV,cAAc,sBAAK,KAAL,C;YA
Cd,IA9kBc,SA8kBv,CAAU,OA AV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;QAhIBP,wB;MA
CJ,C;KARA,C;yFAUA,yB;MAglBA,+C;MAwwEI,0D;MAx1FJ,uC;QAOW,qB;;UAglBO,Q;UAAA,OAAa,SAiwE
X,YAAR,iBAAQ,CAjwEW,CAAb,W;UAAAd,OAAC,cAAAd,C;YAAc,uB;YACV,cAAc,sBAAK,KAAL,C;YACd,IAI
IBc,SAklBV,CAAU,OA AV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;QAplBP,wB;MACJ,C;K
ARA,C;yFAUA,yB;MAolBA,+C;MAkwEI,0D;MAT1FJ,uC;QAOW,qB;;UAolBO,Q;UAAA,OAAa,SA2vEX,YAA
R,iBAAQ,CA3vEW,CAAb,W;UAAAd,OAAC,cAAAd,C;YAAc,uB;YACV,cAAc,sBAAK,KAAL,C;YACd,IAtlBc,SA
slBV,CAAU,OA AV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;QAxlBP,wB;MACJ,C;KARA,C;
yFAUA,yB;MAwlBA,+C;MA4vEI,0D;MAp1FJ,uC;QAOW,qB;;UAwlBO,Q;UAAA,OAAa,SAqvEX,YAAR,iBAA
Q,CArvEW,CAAb,W;UAAAd,OAAC,cAAAd,C;YAAc,uB;YACV,cAAc,sBAAK,KAAL,C;YACd,IA11Bc,SA01BV,C
AAU,OA AV,CAAJ,C;cAAwB,oBAAO,O;cAAP,sB;;;UAE5B,oBAAO,I;;QA5lBP,wB;MACJ,C;KARA,C;mFAU
A,yB;MAAA,8C;MjChJA,6B;MiCgJA,4B;QAQI,OjC9ImC,ciC8IpB,MAAR,iBAAQ,CjC9IoB,C;MiC+IvC,C;KAT
A,C;mFAWA,yB;MAAA,8C;MF5IA,+B;ME4IA,4B;QAQI,OF1IsC,eE0IvB,MAAR,iBAAQ,CF1IuB,C;ME2IIC,C;
KATA,C;mFAWA,yB;MAAA,8C;MICxMA,+B;MkCwMA,4B;QAQI,OICtMsC,ekCsMvB,MAAR,iBAAQ,CICtM
uB,C;MkCuM1C,C;KATA,C;mFAWA,yB;MAAA,8C;MhCtMA,iC;MgCsMA,4B;QAQI,OhCpMyC,gBgCoM1B,
MAAR,iBAAQ,ChCpM0B,C;MgCqM7C,C;KATA,C;mFAWA,yB;MAAA,iE;MAAA,uC;QAQoB,Q;QAAA,2B;Q
AAhB,OAAGb,cAAhB,C;UAAgB,yB;UAAM,IAAI,UAAU,OA AV,CAAJ,C;YAAwB,OAAO,O;;QACrD,MAAM,
gCAAuB,mDAAvB,C;MACV,C;KAVA,C;mFAYA,yB;MAAA,iE;MAAA,uC;QAQoB,Q;QAAA,2B;QAAhB,OA
AgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,UAAU,OA AV,CAAJ,C;YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,
mDAAvB,C;MACV,C;KAVA,C;mFAYA,yB;MAAA,iE;MAAA,uC;QAQoB,Q;QAAA,2B;QAAhB,OAAGb,cAAh
B,C;UAAgB,yB;UAAM,IAAI,UAAU,OA AV,CAAJ,C;YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;
MACV,C;KAVA,C;mFAYA,yB;MAAA,iE;MAAA,uC;QAQoB,Q;QAAA,2B;QAAhB,OAAGb,cAAhB,C;UAAg
B,yB;UAAM,IAAI,UAAU,OA AV,CAAJ,C;YAAwB,OAAO,O;;QACrD,MAAM,gCAAuB,mDAAvB,C;MACV,C;
KAVA,C;;MAkBI,OA AW,mBAAJ,GAAe,IAAf,GAAyB,sBAAK,CAAL,C;IACpC,C;;MAQI,OA AW,mBAAJ,GA
Ae,IAAf,GAAyB,sBAAK,CAAL,C;IACpC,C;;MAQI,OA AW,mBAAJ,GAAe,IAAf,GAAyB,sBAAK,CAAL,C;IA
CpC,C;;MAQI,OA AW,mBAAJ,GAAe,IAAf,GAAyB,sBAAK,CAAL,C;IACpC,C;;MASoB,Q;MAAA,2B;MAAhB,

OAAgB,cAAhB,C;QAAGb,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;;MASoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGb,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;;MASoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGb,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;;MASoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGb,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,O;;MACrD,OAAO,I;IACX,C;2FAEA,yB;MasqfI,8D;MATqfJ,iD;QAOe,oBAAS,C;QAAT,S;UAAc,gBA+pFT,cAAR,iBAAQ,C;;QA/pFhB,OAAO,OAAc,sBAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;MACjE,C;KARA,C;2FAUA,yB;MAoqfI,8D;MAPqfJ,iD;QAOe,oBAAS,C;QAAT,S;UAAc,gBA6pFT,cAAR,iBAAQ,C;;QA7pFhB,OAAO,OAAc,sBAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;MACjE,C;KARA,C;2FAUA,yB;MAkqfI,8D;MAlqfJ,iD;QAOe,oBAAS,C;QAAT,S;UAAc,gBA2pFT,cAAR,iBAAQ,C;;QA3pFhB,OAAO,OAAc,sBAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;MACjE,C;KARA,C;2FAUA,yB;MAGqfI,8D;MAhqfJ,iD;QAOe,oBAAS,C;QAAT,S;UAAc,gBAypFT,cAAR,iBAAQ,C;;QAZpFhB,OAAO,OAAc,sBAAI,KAAJ,CAAtC,GAAsD,aAAa,KAAb,C;MACjE,C;KARA,C;;MAGBe,oBAAS,C;MAAT,S;QAAC,gBAwnFT,gBAAR,iBAAQ,C;;MAXnFhB,OAAO,OAAc,sBAAI,KAAJ,CAAtC,GAAsD,I;IACjE,C;;MAQe,oBAAS,C;MAAT,S;QAAC,gBAunFT,gBAAR,iBAAQ,C;;MAVnFhB,OAAO,OAAc,sBAAI,KAAJ,CAAtC,GAAsD,I;IACjE,C;;MAQe,oBAAS,C;MAAT,S;QAAC,gBAsnFT,gBAAR,iBAAQ,C;;MATnFhB,OAAO,OAAc,sBAAI,KAAJ,CAAtC,GAAsD,I;IACjE,C;;MAQe,oBAAS,C;MAAT,S;QAAC,gBAqnFT,gBAAR,iBAAQ,C;;MARNfhB,OAAO,OAAc,sBAAI,KAAJ,CAAtC,GAAsD,I;IACjE,C;uFAEA,yB;MAAA,kD;MAAA,qC;QAOI,OAAe,QAAR,iBAAQ,EAAQ,OjC1eU,KiC0elB,C;MACnB,C;KARA,C;uFAUA,yB;MAAA,kD;MAAA,qC;QAOI,OAAe,QAAR,iBAAQ,EAAQ,OFzeY,KEyepB,C;MACnB,C;KARA,C;uFAUA,yB;MAAA,kD;MAAA,qC;QAOI,OAAe,QAAR,iBAAQ,EAAQ,OIC1hBY,KkC0hBpB,C;MACnB,C;KARA,C;uFAUA,yB;MAAA,kD;MAAA,qC;QAOI,OAAe,QAAR,iBAAQ,EAAQ,OhCzhBc,KgCyhBtB,C;MACnB,C;KARA,C;iGAUA,yB;MAAA,sC;MjChbA,6B;MiCgbA,0BAOGC,yB;QjCvvhC,6B;;UiCubgC,qB;YAAE,yBjC7aK,ciC6aK,EjC7aL,CiC6aL,C;UAAF,C;QAAA,C;OAAA,C;MAPhC,uC;QAOMb,kBAAR,iB;QAAQ,uB;;UpC2rBf,0D;YACI,IoC5rB0B,UjC7aK,cHymCjB,YAAK,KAAL,CgzmCiB,CiC6aL,CpC4rB1B,C;cACI,sBAAO,K;cAAP,wB;;;UAGR,sBAAO,E;;;QoChsBP,0B;MACJ,C;KARA,C;iGAUA,yB;MAAA,sC;MF3aA,+B;ME2aA,0BAOGC,yB;QFlbhC,+B;;UEkbgC,qB;YAAE,yBFxaQ,eEwaE,EFxaF,C;EwaR,C;UAAF,C;QAAA,C;OAAA,C;MAPhC,uC;QAOMb,kBAAR,iB;QAAQ,uB;;UpC6rBf,0D;YACI,IoC9rB0B,UfXaQ,elCsmCpB,YAAK,KAAL,CkCtmCoB,CEwaR,CpC8rB1B,C;cACI,sBAAO,K;cAAP,wB;;;UAGR,sBAAO,E;;;QoClSBP,0B;MACJ,C;KARA,C;iGAUA,yB;MAAA,sC;MlCteA,+B;MkCseA,0BAOGC,yB;QlC7ehC,+B;;UkC6egC,qB;YAAE,yBICneQ,ekCmeE,ElCneF,CkCmeR,C;UAAF,C;QAAA,C;OAAA,C;MAPhC,uC;QAOMb,kBAAR,iB;QAAQ,uB;;UpC+oBf,0D;YACI,IoChpB0B,UlCneQ,eFmnCpB,YAAK,KAAL,CEnnCoB,CkCmeR,CpCgpB1B,C;cACI,sBAAO,K;cAAP,wB;;;UAGR,sBAAO,E;;;QoCpBP,0B;MACJ,C;KARA,C;iGAUA,yB;MAAA,sC;MhCneA,iC;MgCmeA,0BAOGC,yB;QhC1ehC,iC;;UgC0egC,qB;YAAE,yBhCheW,gBgCgeD,EhCheC,CgCgeX,C;UAAF,C;QAAA,C;OAAA,C;MAPhC,uC;QAOMb,kBAAR,iB;QAAQ,uB;;UpCipBf,0D;YACI,IoClpB0B,UhCheW,gBJknCvB,YAAK,KAAL,CllnCuB,CgCgeX,CpCkpB1B,C;cACI,sBAAO,K;cAAP,wB;;;UAGR,sBAAO,E;;;QoCtpBP,0B;MACJ,C;KARA,C;+FAUA,yB;MAAA,sC;MpCkwBA,0D;MAAA,+C;MG1tCA,6B;MiCwdA,yBAO+B,yB;QjC/d/B,6B;;UiC+d+B,qB;YAAE,yBjCrdM,ciCqdI,EjCrdJ,CiCqdN,C;UAAF,C;QAAA,C;OAAA,C;MAP/B,uC;QAOMb,kBAAR,iB;QAAQ,sB;;UpC+vBD,Q;UAAA,OAAQ,SAAR,wBAAQ,CAAR,W;UAAAd,OAAc,cAAAd,C;YAAc,uB;YACV,IoChwByB,UjCrdM,cHqtCjB,YAAK,KAAL,CGrtCiB,CiCqdN,CpCgwBzB,C;cACI,qBAAO,K;cAAP,uB;;;UAGR,qBAAO,E;;;QoCpwBP,yB;MACJ,C;KARA,C;+FAUA,yB;MAAA,sC;MpCowBA,0D;MAAA,+C;MkCvtCA,+B;MEmdA,yBAO+B,yB;QF1d/B,+B;;UE0d+B,qB;YAAE,yBFhdS,eEgdC,EFhdD,CEgdT,C;UAAF,C;QAAA,C;OAAA,C;MAP/B,uC;QAOMb,kBAAR,iB;QAAQ,sB;;UpCiwBD,Q;UAAA,OAAQ,SAAR,wBAAQ,CAAR,W;UAAAd,OAAc,cAAAd,C;YAAc,uB;YACV,IoClwByB,UfhdS,elCktCpB,YAAK,KAAL,CkClCoB,CEgdT,CpCkwBzB,C;cACI,qBAAO,K;cAAP,uB;;;UAGR,qBAAO,E;;;QoCtwBP,yB;MACJ,C;KARA,C;+FAUA,yB;MAAA,sC;MpCstBA,0D;MAAA,+C;MEpuCA,+B;MkC8gBA,yBAO+B,yB;QlCrhB/B,+B;;UkCqhB+B,qB;YAAE,yBIC3gBS,ekC2gBC,EIC3gBD,CkC2gBT,C;UAAF,C;QAAA,C;OAAA,C;MAP/B,uC;QAOMb,kBAAR,iB;QAAQ,sB;;UpCmtBD,Q;UAAA,OAAQ,SAAR,wBAAQ,CAAR,W;UAAAd,OAAc,cAAAd,C;YAAc,uB;YACV,IoCptByB,UIC3gBS,eF+tCpB,YAAK,KAAL,CE/tCoB,CkC2gBT,CpCotBzB,C;cACI,qBAAO,K;cAAP,uB;;;UAGR,qBAAO,E;;;QoCxtBP,yB;MACJ,C;KARA,C;+FAUA,yB;MAAA,sC;MpCwtBA,0D;MAAA,+C;MInuCA,iC;MgC2gBA,yBAO+B,yB;

QhClhB/B,iC;;UgCkhB+B,qB;YAAE,yBhCxBY,gBgCwgBF,EhCxBE,CgCwgBZ,C;UAAF,C;QAAA,C;OAAA,
C;MAP/B,uC;QAOMb,kBAAR,iB;QAAQ,sB;;UpCqtBD,Q;UAAA,OAAQ,SAAR,wBAAQ,CAAR,W;UAAAd,OAA
c,cAAAd,C;YAAc,uB;YACV,IoCttByB,UhCxBY,gBJ8tCvB,YAAK,KAAL,CI9tCuB,CgCwgBZ,CpCstBzB,C;cAC
I,qBAAO,K;cAAP,uB;;UAGR,qBAAO,E;;;QoC1tBP,yB;MACJ,C;KARA,C;iFAUA,yB;MAAA,4C;MjChgBA,6B
;MiCggBA,4B;QAQI,OjC9fmC,ciC8fpB,KAAR,iBAAQ,CjC9foB,C;MiC+fvC,C;KATA,C;iFAWA,yB;MAAA,4C;
MF5fA,+B;ME4fA,4B;QAQI,OF1fsC,eE0fvB,KAAR,iBAAQ,CF1fuB,C;ME2f1C,C;KATA,C;iFAWA,yB;MAAA,
4C;MICxjBA,+B;MkCwjBA,4B;QAQI,OICtjBsC,ekCsjBvB,KAAR,iBAAQ,CICtjBuB,C;MkCujB1C,C;KATA,C;i
FAWA,yB;MAAA,4C;MhCtjBA,iC;MgCsjBA,4B;QAQI,OhCpjByC,gBgCojB1B,KAAR,iBAAQ,ChCpjB0B,C;M
gCqjB7C,C;KATA,C;iFAWA,yB;MAAA,+C;MAAA,iE;MA5EI,0D;MA5EJ,uC;QAQkB,Q;QAAA,OAAa,SA84
EX,YA94EF,SA84EN,QAAQ,CA94EW,CAAb,W;QAAd,OAAc,cAAAd,C;UAAc,uB;UACV,cAAc,sBAAK,KAAL,
C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAb
A,C;iFAeA,yB;MAAA,+C;MAAA,iE;MA+4EI,0D;MA/4EJ,uC;QAQkB,Q;QAAA,OAAa,SAu4EX,YAv4EF,SAu4
EN,QAAQ,CAv4EW,CAAb,W;QAAd,OAAc,cAAAd,C;UAAc,uB;UACV,cAAc,sBAAK,KAAL,C;UACd,IAAI,UA
AU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAbA,C;iFAeA,yB;M
AAA,+C;MAAA,iE;MAw4EI,0D;MAx4EJ,uC;QAQkB,Q;QAAA,OAAa,SAg4EX,YAh4EF,SAg4EN,QAAQ,CAh
4EW,CAAb,W;QAAd,OAAc,cAAAd,C;UAAc,uB;UACV,cAAc,sBAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAA
J,C;YAAwB,OAAO,O;;QAEEnC,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAbA,C;iFAeA,yB;MAAA,+C;MAAA,
iE;MAi4EI,0D;MAj4EJ,uC;QAQkB,Q;QAAA,OAAa,SAy3EX,YAz3EF,SAy3EN,QAAQ,CAz3EW,CAAb,W;QA
Ad,OAAc,cAAAd,C;UAAc,uB;UACV,cAAc,sBAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAA
O,O;;QAEEnC,MAAM,gCAAuB,mDAAvB,C;MACV,C;KAbA,C;+FAeA,yB;MAAA,0D;MAAA,qC;QAQI,OAAe,
YAAR,iBAAQ,EAAY,OjC1sBM,KiC0sBIB,C;MACnB,C;KARA,C;+FAUA,yB;MAAA,0D;MAAA,qC;QAQI,OA
Ae,YAAR,iBAAQ,EAAY,OFzsBQ,KEysBpB,C;MACnB,C;KARA,C;+FAUA,yB;MAAA,0D;MAAA,qC;QAQI,O
AAe,YAAR,iBAAQ,EAAY,OIC1vBQ,KkC0vBpB,C;MACnB,C;KARA,C;+FAUA,yB;MAAA,0D;MAAA,qC;QA
OI,OAAe,YAAR,iBAAQ,EAAY,OhCzvBU,KgCyvBtB,C;MACnB,C;KARA,C;;MAGBI,OAAW,mBAAJ,GAAe,I
AAf,GAAyB,sBAAK,iBAAO,CAAP,IAAL,C;IACpC,C;;MAQI,OAAW,mBAAJ,GAAe,IAAf,GAAyB,sBAAK,iB
AAO,CAAP,IAAL,C;IACpC,C;;MAQI,OAAW,mBAAJ,GAAe,IAAf,GAAyB,sBAAK,iBAAO,CAAP,IAAL,C;IA
CpC,C;;MAQI,OAAW,mBAAJ,GAAe,IAAf,GAAyB,sBAAK,iBAAO,CAAP,IAAL,C;IACpC,C;6FAEA,yB;MAA
A,+C;MA8wEI,0D;MA9wEJ,uC;QAQkB,Q;QAAA,OAAa,SAuwEX,YAvwEF,SAuwEN,QAAQ,CAvwEW,CAAb,
W;QAAd,OAAc,cAAAd,C;UAAc,uB;UACV,cAAc,sBAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB
,OAAO,O;;QAEEnC,OAAO,I;MACX,C;KAZA,C;6FAcA,yB;MAAA,+C;MAwwEI,0D;MAxwEJ,uC;QAQkB,Q;Q
AAA,OAAa,SAiwEX,YAjwEF,SAiwEN,QAAQ,CAjwEW,CAAb,W;QAAd,OAAc,cAAAd,C;UAAc,uB;UACV,cA
Ac,sBAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,OAAO,I;MACX,C;KAZA,
C;6FAcA,yB;MAAA,+C;MAkwEI,0D;MAlwEJ,uC;QAQkB,Q;QAAA,OAAa,SA2vEX,YA3vEF,SA2vEN,QAAQ,
CA3vEW,CAAb,W;QAAd,OAAc,cAAAd,C;UAAc,uB;UACV,cAAc,sBAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,
CAAJ,C;YAAwB,OAAO,O;;QAEEnC,OAAO,I;MACX,C;KAZA,C;6FAcA,yB;MAAA,+C;MA4vEI,0D;MA5vEJ,u
C;QAQkB,Q;QAAA,OAAa,SAqvEX,YArvEF,SAqvEN,QAAQ,CArvEW,CAAb,W;QAAd,OAAc,cAAAd,C;UAAc,
uB;UACV,cAAc,sBAAK,KAAL,C;UACd,IAAI,UAAU,OAAV,CAAJ,C;YAAwB,OAAO,O;;QAEEnC,OAAO,I;MA
CX,C;KAZA,C;qFAcA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QASI,OAAO,kBAAO,cAAP,C;MACX,C;KAVA,C
;qFAYA,yB;MAAA,mC;MAAA,gD;MAAA,4B;QASI,OAAO,kBAAO,cAAP,C;MACX,C;KAVA,C;qFAYA,yB;
MAAA,mC;MAAA,gD;MAAA,4B;QASI,OAAO,kBAAO,cAAP,C;MACX,C;KAVA,C;qFAYA,yB;MAAA,mC;M
AAA,gD;MAAA,4B;QASI,OAAO,kBAAO,cAAP,C;MACX,C;KAVA,C;;MAoBI,IAAI,mBAAJ,C;QACI,MAAM,
2BAAuB,iBAAvB,C;MACV,OAAO,sBAAI,MAAO,iBAAQ,cAAR,CAAX,C;IACX,C;;MAUI,IAAI,mBAAJ,C;Q
ACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,sBAAI,MAAO,iBAAQ,cAAR,CAAX,C;IACX,C;;MAUI,IAAI,
mBAAJ,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,sBAAI,MAAO,iBAAQ,cAAR,CAAX,C;IACX,C;;
MAUI,IAAI,mBAAJ,C;QACI,MAAM,2BAAuB,iBAAvB,C;MACV,OAAO,sBAAI,MAAO,iBAAQ,cAAR,CAAX,
C;IACX,C;qFAEA,yB;MAAA,gD;MjCh1BA,6B;MiCg1BA,4B;QAQI,OjC70BmC,ciC60BpB,OAAR,iBAAQ,CjC
70BoB,C;MiC80BvC,C;KARA,C;qFAUA,yB;MAAA,gD;MF30BA,+B;ME20BA,4B;QAQI,OFx0BsC,eEw0BvB,
OAAR,iBAAQ,CFx0BuB,C;MEy0B1C,C;KARA,C;qFAUA,yB;MAAA,gD;MICt4BA,+B;MkCs4BA,4B;QAQI,OI

Cn4BsC,ekCm4BvB,OAAR,iBAAQ,CiCn4BuB,C;MkCo4B1C,C;KARA,C;qFAUA,yB;MAAA,gD;MhCn4BA,iC;MgCm4BA,4B;QAOI,OhCh4ByC,gBgCg4B1B,OAAR,iBAAQ,ChCh4B0B,C;MgCi4B7C,C;KARA,C;qFAUA,yB;MAAA,kF;MAAA,iE;MAAA,wB;MAAA,8B;MAAA,uC;QASoB,UAST,M;QAXP,aAAoB,I;QACpB,YAA Y,K;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,gDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAA Y,MAAM,gCAAuB,mDAAvB,C;QAEIB,OAAO,0D;MACX,C;KAnBA,C;qFAqBA,yB;MAAA,kF;MAAA,iE;MAAA,0B;MAAA,8B;MAAA,uC;QASoB,UAST,M;QAXP,aAAqB,I;QACrB,YAA Y,K;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,gDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAA Y,MAAM,gCAAuB,mDAAvB,C;QAEIB,OAAO,2D;MACX,C;KAnBA,C;qFAqBA,yB;MAAA,kF;MAAA,iE;MAAA,0B;MAAA,8B;MAAA,uC;QASoB,UAST,M;QAXP,aAAqB,I;QACrB,YAA Y,K;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,gDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAA Y,MAAM,gCAAuB,mDAAvB,C;QAEIB,OAAO,2D;MACX,C;KAnBA,C;qFAqBA,yB;MAAA,kF;MAAA,iE;MAAA,4B;MAAA,8B;MAAA,uC;QASoB,UAST,M;QAXP,aAA sB,I;QACtB,YAA Y,K;QACI,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,UAAU,OAAV,CAAJ,C;YACI,IAAI,KAAJ,C;cAAW,MAAM,8BAAyB,gDAAzB,C;YACjB,SAAS,O;YACT,QAAQ,I;;;QAGhB,IAAI,CAAC,KAAL,C;UAA Y,MAAM,gCAAuB,mDAAvB,C;QAEIB,OAAO,4D;MACX,C;KAnBA,C;;MA2BI,OAAW,mBAAQ,CAAZ,GAAe,sBA AK,CAAL,CAAf,GAA4B,I;IACvC,C;;MAQI,OAAW,mBAAQ,CAAZ,GAAe,sBA AK,CAAL,CAAf,GAA4B,I;IACvC,C;;MAQI,OAAW,mBAAQ,CAAZ,GAAe,sBA AK,CAAL,CAAf,GAA4B,I;IACvC,C;;MAW oB,Q;MAFhB,aAAoB,I;MACpB,YAA Y,K;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAA W,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAA Y,OAAO,I;MACnB,OAAO,M;IACX,C;;MAW oB,Q;MAFhB,aAAqB,I;MACrB,YAA Y,K;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAA W,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAA Y,OAAO,I;MACnB,OAAO,M;IACX,C;;MAW oB,Q;MAFhB,aAA sB,I;MACTb,YAA Y,K;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAA W,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAA Y,OAAO,I;MACnB,OAAO,M;IACX,C;;MAW oB,Q;MAFhB,aAA sB,I;MACTb,YAA Y,K;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,UAAU,OAAV,CAAJ,C;UACI,IAAI,KAAJ,C;YAA W,OAAO,I;UACIB,SAAS,O;UACT,QAAQ,I;;;MAGhB,IAAI,CAAC,KAAL,C;QAA Y,OAAO,I;MACnB,OAAO,M;IACX,C;;MvB3zCI,IAAI,EUbu0CI,KAAK,CvBv0CT,CAAJ,C;QACI,cuBs0Cc,sD;QvBr0Cd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuBs0CV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAC,CAAd,CAApB,C;IACX,C;;MvBz0CI,IAAI,EUbq1CI,KAAK,CvBr1CT,CAAJ,C;QACI,cuBo1Cc,sD;QvBn1Cd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuBo1CV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAC,CAAd,CAApB,C;IACX,C;;MvBv1CI,IAAI,EUbm2CI,KAAK,CvBn2CT,CAAJ,C;QACI,cuBk2Cc,sD;QvBj2Cd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuBk2CV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAC,CAAd,CAApB,C;IACX,C;;MvBr2CI,IAAI,EUbi3CI,KAAK,CvBj3CT,CAAJ,C;QACI,cuBg3Cc,sD;QvB/2Cd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuBg3CV,OAAO,uBAAoB,gBAAV,iBAAO,CAAP,IAAU,EAAC,CAAd,CAApB,C;IACX,C;;MvBn3CI,IAAI,EUb+3CI,KAAK,CvB/3CT,CAAJ,C;QACI,cuB83Cc,sD;QvB73Cd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuB83CV,OAAO,mBAAgB,gBAAV,iBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB,C;IACX,C;;MvBj4CI,IAAI,EUb64CI,KAAK,CvB74CT,CAAJ,C;QACI,cuB44Cc,sD;QvB34Cd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuB44CV,OAAO,mBAAgB,gBAAV,iBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB,C;IACX,C;;MvB/4CI,IAAI,EUb25CI,KAAK,CvB35CT,CAAJ,C;QACI,cuB05Cc,sD;QvBz5Cd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuB05CV,OAAO,mBAAgB,gBAAV,iBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB,C;IACX,C;;MvB75CI,IAAI,EUBy6CI,KAAK,CvBz6CT,CAAJ,C;QACI,cuBw6Cc,sD;QvBv6Cd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuBw6CV,OAAO,mBAAgB,gBAAV,iBAAO,CAAP,IAAU,EAAC,CAAd,CAAhB,C;IACX,C;mGAEA,yB;MAAA,4C;MAAA,qD;MAkzDI,8D;MAIzDJ,uC;QASI,iBAyyDgB,cAAR,iBAAQ,CAzyDhB,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,W;MACX,C;KafA,C;mGAiBA,yB;MAAA,4C;MAAA,qD;MAyyDI,8D;MAzyDJ,uC;QASI,iBAgyDg

B,cAAR,iBAAQ,CAhyDhB,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YAC I,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,W;MACX,C;KafA,C;mGaiBA,yB;MAAA,4C;MAAA,q D;MAgyDI,8D;MAhyDJ,uC;QASI,iBAuxDgB,cAAR,iBAAQ,CAvxDhB,WAA+B,CAA/B,U;UACI,IAAI,CAAC, UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;;QAGf,OAAO,W;MACX, C;KafA,C;mGaiBA,yB;MAAA,4C;MAAA,qD;MAuxDI,8D;MAvxDJ,uC;QASI,iBA8wDgB,cAAR,iBAAQ,CA9 wDhB,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QA AQ,CAAR,IAAL,C;;;QAGf,OAAO,W;MACX,C;KafA,C;2FAiBA,yB;MAAA,+D;MAAA,uC;QAWiB,Q;QAFb,e AAe,K;QACf,WAAW,gB;QACE,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,IAAI,QAAJ,C;YACI,IAAK,WAAI,I AAJ,C;eACJ,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,IAAK,WAAI,IAAJ,C;YACL,WAAW,I;;;QAEEnB,OAA O,I;MACX,C;KAnBA,C;2FAqBA,yB;MAAA,+D;MAAA,uC;QAWiB,Q;QAFb,eAAe,K;QACf,WAAW,gB;QACE ,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,IAAI,QAAJ,C;YACI,IAAK,WAAI,IAAJ,C;eACJ,IAAI,CAAC,UAA U,IAAV,CAAL,C;YACD,IAAK,WAAI,IAAJ,C;YACL,WAAW,I;;;QAEEnB,OAAO,I;MACX,C;KAnBA,C;2FAqB A,yB;MAAA,+D;MAAA,uC;QAWiB,Q;QAFb,eAAe,K;QACf,WAAW,gB;QACE,2B;QAAb,OAAa,cAAb,C;UAA a,sB;UACT,IAAI,QAAJ,C;YACI,IAAK,WAAI,IAAJ,C;eACJ,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,IAAK, WAAI,IAAJ,C;YACL,WAAW,I;;;QAEEnB,OAAO,I;MACX,C;KAnBA,C;2FAqBA,yB;MAAA,+D;MAAA,uC;QA WiB,Q;QAFb,eAAe,K;QACf,WAAW,gB;QACE,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,IAAI,QAAJ,C;YACI ,IAAK,WAAI,IAAJ,C;eACJ,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACD,IAAK,WAAI,IAAJ,C;YACL,WAAW,I;; ;QAEEnB,OAAO,I;MACX,C;KAnBA,C;qFAqBA,yB;MAAA,+D;MAAA,uC;QAOW,kBAAS,gB;QAwOA,Q;QAA A,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAxOa,SAwOT,CAAU,OAAV,CAAJ,C;YAAwB,WAAy,W AAI,OAAJ,C;;QAxO1D,OAYOO,W;MAxOX,C;KARA,C;qFAUA,yB;MAAA,+D;MAAA,uC;QAOW,kBAAS,gB; QAYOA,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAzOc,SAyOV,CAAU,OAAV,CAAJ,C;YA AwB,WAAy,WAAI,OAAJ,C;;QAzO1D,OA0OO,W;MAzOX,C;KARA,C;qFAUA,yB;MAAA,+D;MAAA,uC;QA OW,kBAAS,gB;QA0OA,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IA1Oc,SA0OV,CAAU,OA AV,CAAJ,C;YAAwB,WAAy,WAAI,OAAJ,C;;QA1O1D,OA2OO,W;MA1OX,C;KARA,C;qFAUA,yB;MAAA,+ D;MAAA,uC;QAOW,kBAAS,gB;QA2OA,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IA3Oe,S A2OX,CAAU,OAAV,CAAJ,C;YAAwB,WAAy,WAAI,OAAJ,C;;QA3O1D,OA4OO,W;MA3OX,C;KARA,C;kGA UA,yB;MAAA,+D;MAAA,uC;QASW,kBAAGB,gB;QAqkGV,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb ,C;UAAa,sB;UaphGT,IAjDsC,SAiDIC,EAohGkB,cAphGIB,EAohGkB,sBAphGIB,WAohG2B,IAphG3B,CAAJ,C; YAA2C,sBAohGZ,IAphGY,C;;QAjD/C,OAmDO,W;MAIDX,C;KAVA,C;mGAYA,yB;MAAA,+D;MAAA,uC;QA SW,kBAAGB,gB;QaskGV,gB;QADb,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UAlhGT,IApDuC,SA oDnC,EAKhGkB,cAlhGIB,EAKhGkB,sBALhGIB,WAKhG2B,IALhG3B,CAAJ,C;YAA2C,sBAkhGZ,IALhGY,C;;QA pD/C,OAsDO,W;MARDX,C;KAVA,C;mGAYA,yB;MAAA,+D;MAAA,uC;QASW,kBAAGB,gB;QAukGV,gB;QA Db,YAAY,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UAhhGT,IAvDuC,SAuDnC,EAghGkB,cAhhGIB,EAgh GkB,sBAhhGIB,WAghG2B,IAhhG3B,CAAJ,C;YAA2C,sBAghGZ,IAhhGY,C;;QAvD/C,OAYDO,W;MAxDX,C;K AVA,C;mGAYA,yB;MAAA,+D;MAAA,uC;QASW,kBAAGB,gB;QAwkGV,gB;QADb,YAAY,C;QACC,2B;QAA b,OAAa,cAAb,C;UAAa,sB;UA9gGT,IA1DwC,SA0DpC,EA8gGkB,cA9gGIB,EA8gGkB,sBA9gGIB,WA8gG2B,I A9gG3B,CAAJ,C;YAA2C,sBA8gGZ,IA9gGY,C;;QA1D/C,OA4DO,W;MA3DX,C;KAVA,C;;MA0iGiB,gB;MAD b,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QAphGT,IAAI,WAohGkB,cAphGIB,EAohGkB,sBAphGI B,WAohG2B,IAphG3B,CAAJ,C;UAA2C,sBAohGZ,IAphGY,C;;MAE/C,OAAO,W;IACX,C;;MA8hGiB,gB;MAD b,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QAlhGT,IAAI,WAKhGkB,cAlhGIB,EAKhGkB,sBALhGIB ,WAKhG2B,IALhG3B,CAAJ,C;UAA2C,sBAkhGZ,IALhGY,C;;MAE/C,OAAO,W;IACX,C;;MA4hGiB,gB;MADb, YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QAhhGT,IAAI,WAghGkB,cAhhGIB,EAghGkB,sBAhhGIB ,WAghG2B,IAhhG3B,CAAJ,C;UAA2C,sBAghGZ,IAhhGY,C;;MAE/C,OAAO,W;IACX,C;;MA0hGiB,gB;MADb, YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QA9gGT,IAAI,WA8gGkB,cA9gGIB,EA8gGkB,sBA9gGIB ,WA8gG2B,IA9gG3B,CAAJ,C;UAA2C,sBA8gGZ,IA9gGY,C;;MAE/C,OAAO,W;IACX,C;2FAEA,yB;MAAA,+D ;MAAA,uC;QAOW,kBAAY,gB;QAwCH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CA xCY,SAwCX,CAAU,OAAV,CAAL,C;YAAyB,WAAy,WAAI,OAAJ,C;;QAxC3D,OAYCO,W;MAxCX,C;KARA, C;2FAUA,yB;MAAA,+D;MAAA,uC;QAOW,kBAAY,gB;QAYCH,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAA

gB,yB;UAAM,IAAI,CAzCa,SAyCZ,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAAJ,C;;QAzC3D,OA0CO,
W;MAzCX,C;KARA,C;2FAUA,yB;MAAA,+D;MAAA,uC;QAOW,kBAAY,gB;QA0CH,Q;QAAA,2B;QAAhB,O
AAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CA1Ca,SA0CZ,CAAU,OAAV,CAAL,C;YAAyB,WAAY,WAAI,OAA
J,C;;QA1C3D,OA2CO,W;MA1CX,C;KARA,C;2FAUA,yB;MAAA,+D;MAAA,uC;QAOW,kBAAY,gB;QA2CH,Q
;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,CA3Cc,SA2Cb,CAAU,OAAV,CAAL,C;YAAy
B,WAAY,WAAI,OAAJ,C;;QA3C3D,OA4CO,W;MA3CX,C;KARA,C;;MAiBoB,Q;MAAA,2B;MAAhB,OAAgB,c
AAhB,C;QAAgB,yB;QAAM,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,O
AAO,W;IACX,C;;MASoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,CAAC,UAAU,OA
AV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;IACX,C;;MASoB,Q;MAAA,2B;MAAhB,OA
AgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC
3D,OAAO,W;IACX,C;;MASoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,CAAC,UAA
U,OAAV,CAAL,C;UAAyB,WAAY,WAAI,OAAJ,C;;MAC3D,OAAO,W;IACX,C;;MASoB,Q;MAAA,2B;MAAh
B,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,WAAY,WAAI,OAAJ,C;;MAC1
D,OAAO,W;IACX,C;;MASoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAA
V,CAAJ,C;UAAwB,WAAY,WAAI,OAAJ,C;;MAC1D,OAAO,W;IACX,C;;MASoB,Q;MAAA,2B;MAAhB,OAAg
B,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,WAAY,WAAI,OAAJ,C;;MAC1D,OAAO
,W;IACX,C;;MASoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C
;UAAwB,WAAY,WAAI,OAAJ,C;;MAC1D,OAAO,W;IACX,C;;MAQI,IAAI,OAAQ,UAAZ,C;QAAuB,O9BjzDe,
W;;M8BkzDtC,OAA4D,SAmvCrD,cAAkB,cAAR,iBAAQ,EAnvCN,OAAQ,MAmvCF,EAnvCS,OAAQ,aAAR,GA
AuB,CAAvB,IAMvCT,CAAIb,CAnvCqD,C;IACHe,C;;MAQI,IAAI,OAAQ,UAAZ,C;QAAuB,O9B3zDe,W;;M8B
4zDtC,OAA4D,SASvCrD,eAAmB,cAAR,iBAAQ,EAtvCP,OAAQ,MAsvCD,EAtvCQ,OAAQ,aAAR,GAAuB,CAA
vB,IAsvCR,CAAnB,CAtvCqD,C;IACHe,C;;MAQI,IAAI,OAAQ,UAAZ,C;QAAuB,O9Br0De,W;;M8Bs0DtC,OAA
4D,UAYvCrD,eAAmB,cAAR,iBAAQ,EAzvCP,OAAQ,MAyvCD,EAzvCQ,OAAQ,aAAR,GAAuB,CAAvB,IAYvC
R,CAAnB,CAzvCqD,C;IACHe,C;;MAQI,IAAI,OAAQ,UAAZ,C;QAAuB,O9B/0De,W;;M8Bg1DtC,OAA4D,UA4v
CrD,gBAAoB,cAAR,iBAAQ,EA5vCR,OAAQ,MA4vCA,EA5vCO,OAAQ,aAAR,GAAuB,CAAvB,IA4vCP,CAAp
B,CA5vCqD,C;IACHe,C;;MAWkB,Q;MAHd,WAAmB,wBAAR,OAAQ,EAawB,EAxB,C;MACnB,IAAI,SAAQ
,CAAZ,C;QAAe,OAAO,W;MACTb,WAAW,iBAAGB,IAAhB,C;MACG,yB;MAAd,OAAC,cAAAd,C;QAAc,uB;QA
CV,IAAK,WAAI,sBAAI,KAAJ,CAAJ,C;;MAET,OAAO,I;IACX,C;;MAWkB,Q;MAHd,WAAmB,wBAAR,OAAQ
,EAawB,EAxB,C;MACnB,IAAI,SAAQ,CAAZ,C;QAAe,OAAO,W;MACTb,WAAW,iBAAiB,IAAjB,C;MACG,
yB;MAAd,OAAC,cAAAd,C;QAAc,uB;QACV,IAAK,WAAI,sBAAI,KAAJ,CAAJ,C;;MAET,OAAO,I;IACX,C;;MA
WkB,Q;MAHd,WAAmB,wBAAR,OAAQ,EAawB,EAxB,C;MACnB,IAAI,SAAQ,CAAZ,C;QAAe,OAAO,W;M
ACTb,WAAW,iBAAiB,IAAjB,C;MACG,yB;MAAd,OAAC,cAAAd,C;QAAc,uB;QACV,IAAK,WAAI,sBAAI,KAAJ
,CAAJ,C;;MAET,OAAO,I;IACX,C;;MAWkB,Q;MAHd,WAAmB,wBAAR,OAAQ,EAawB,EAxB,C;MACnB,I
AAI,SAAQ,CAAZ,C;QAAe,OAAO,W;MACTb,WAAW,iBAaKB,IAAIb,C;MACG,yB;MAAd,OAAC,cAAAd,C;QA
Ac,uB;QACV,IAAK,WAAI,sBAAI,KAAJ,CAAJ,C;;MAET,OAAO,I;IACX,C;;MAQI,OAAO,cAAkB,aAAR,iBA
AQ,EAawB,OAAX,CAAIb,C;IACX,C;;MAQI,OAAO,eAAmB,aAAR,iBAAQ,EAawB,OAAX,CAAnB,C;IACX,C;
;MAQI,OAAO,eAAmB,aAAR,iBAAQ,EAawB,OAAX,CAAnB,C;IACX,C;;MAQI,OAAO,gBAAoB,aAAR,iBAA
Q,EAawB,OAAX,CAApB,C;IACX,C;;MAQI,OAAO,cAAkB,cAAR,iBAAQ,EAawB,OAAX,CAAIb,C;IACX,C;;M
AQI,OAAO,eAAmB,cAAR,iBAAQ,EAawB,OAAX,CAAnB,C;IACX,C;;MAQI,OAAO,eAAmB,aAAR,iBAAQ,E
AAW,OAAX,CAAnB,C;IACX,C;;MAQI,OAAO,gBAAoB,cAAR,iBAAQ,EAawB,OAAX,CAApB,C;IACX,C;;M
AkBiB,UACL,M;MvBthER,IAAI,EuB+gEI,KAAK,CvB/gET,CAAJ,C;QACI,cuB8gEc,sD;QvB7gEd,MAAM,gCA
AyB,OAAQ,WAAjC,C;;MuB8gEV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,cAAT,C;QA
Ae,OAAO,mB;MACTb,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C
;MACZ,WAAW,iBAAGB,CAAhB,C;MACE,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAI,qDAAW,CAAF,C;
UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;MAET,OAAO,I;IACX,C;;MAkBiB,UACL,M;MvB9iER,IAAI,EuBuiEI,KA
AK,CvBviET,CAAJ,C;QACI,cuBsiEc,sD;QvBriEd,MAAM,gCAAYB,OAAQ,WAAjC,C;;MuBsiEV,IAAI,MAAK,
CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,cAAT,C;QAAe,OAAO,mB;MACTb,IAAI,MAAK,CAAT,C;QA
AY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAiB,CAAJB,C;MACE,2B;M

AAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAI,qDAAW,CAAf,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;MAET,OAA
O,I;IACX,C;;MAkBiB,UACL,M;MvBtkER,IAAI,EUb+jEI,KAAK,CvB/jET,CAAJ,C;QACI,cuB8jEc,sD;QvB7jEd,
MAAM,gCAAyB,OAAQ,WAAjC,C;;MuB8jEV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,c
AAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACn
B,YAAY,C;MACZ,WAAW,iBAAiB,CAAjB,C;MACE,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAI,qDAA
W,CAAf,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;;MAET,OAAO,I;IACX,C;;MAkBiB,UACL,M;MvB9IER,IAAI,
EuBulEI,KAAK,CvBvIET,CAAJ,C;QACI,cuBslEc,sD;QvBrlEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuBslEV,I
AAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,IAAI,KAAK,cAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,
CAAT,C;QAAY,OAAO,OAAO,sBAAK,CAAL,CAAP,C;MACnB,YAAY,C;MACZ,WAAW,iBAAkB,CAAIB,C;
MACE,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,IAAI,qDAAW,CAAf,C;UACI,K;QACJ,IAAK,WAAI,IAAJ,C;
;MAET,OAAO,I;IACX,C;;MvBnmEI,IAAI,EUb+mEI,KAAK,CvB/mET,CAAJ,C;QACI,cuB8mEc,sD;QvB7mEd,
MAAM,gCAAyB,OAAQ,WAAjC,C;;MuB8mEV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,c;M
ACX,IAAI,KAAK,IAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,OA
AO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAgB,CAAhB,C;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IA
A7B,U;QACI,IAAK,WAAI,sBAAK,KAAL,CAAJ,C;MACT,OAAO,I;IACX,C;;MvBxnEI,IAAI,EUBooEI,KAAK,
CvBpoET,CAAJ,C;QACI,cuBmoEc,sD;QvBloEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuBmoEV,IAAI,MAAK,
CAAT,C;QAAY,OAAO,W;MACnB,WAAW,c;MACX,IAAI,KAAK,IAAT,C;QAAe,OAAO,mB;MACtB,IAAI,M
AAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,OAAO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAiB,CAAjB,C
;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,sBAAK,KAAL,CAAJ,C;MACT,OA
AO,I;IACX,C;;MvB7oEI,IAAI,EUBypEI,KAAK,CvBzpET,CAAJ,C;QACI,cuBwpEc,sD;QvBvpEd,MAAM,gCAA
yB,OAAQ,WAAjC,C;;MuBwpEV,IAAI,MAAK,CAAT,C;QAAY,OAAO,W;MACnB,WAAW,c;MACX,IAAI,KA
AK,IAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QAAY,OAAO,OAAO,sBAAK,OAAO,CAAP,IAA
L,CAAP,C;MACnB,WAAW,iBAAiB,CAAjB,C;MACX,iBAAc,OAAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IA
AK,WAAI,sBAAK,KAAL,CAAJ,C;MACT,OAAO,I;IACX,C;;MvBlqEI,IAAI,EUb8qEI,KAAK,CvB9qET,CAAJ,
C;QACI,cuB6qEc,sD;QvB5qEd,MAAM,gCAAyB,OAAQ,WAAjC,C;;MuB6qEV,IAAI,MAAK,CAAT,C;QAAY,O
AAO,W;MACnB,WAAW,c;MACX,IAAI,KAAK,IAAT,C;QAAe,OAAO,mB;MACtB,IAAI,MAAK,CAAT,C;QA
AY,OAAO,OAAO,sBAAK,OAAO,CAAP,IAAL,CAAP,C;MACnB,WAAW,iBAAkB,CAAIB,C;MACX,iBAAc,O
AAO,CAAP,IAAd,UAA6B,IAA7B,U;QACI,IAAK,WAAI,sBAAK,KAAL,CAAJ,C;MACT,OAAO,I;IACX,C;mG
AEA,yB;MAAA,4C;MAAA,gD;MA6hCI,8D;MA7hCJ,uC;QASI,iBA6hCgB,cAAR,iBAAQ,CA7hChB,WAA+B,CA
A/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C
;;QAGf,OAAO,iB;MACX,C;KafA,C;mGAiBA,yB;MAAA,4C;MAAA,gD;MA6hCI,8D;MA7hCJ,uC;QASI,iBAo
hCgB,cAAR,iBAAQ,CaphChB,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;
YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;QAGf,OAAO,iB;MACX,C;KafA,C;mGAiBA,yB;MAAA,4C;MA
AA,gD;MAohCI,8D;MAphCJ,uC;QASI,iBA2gCgB,cAAR,iBAAQ,CA3gChB,WAA+B,CAA/B,U;UACI,IAAI,CA
AC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,QAAQ,CAAR,IAAL,C;;QAGf,OAAO,iB;MA
CX,C;KafA,C;mGAiBA,yB;MAAA,4C;MAAA,gD;MA2gCI,8D;MA3gCJ,uC;QASI,iBAkgCgB,cAAR,iBAAQ,C
AlgChB,WAA+B,CAA/B,U;UACI,IAAI,CAAC,UAAU,sBAAK,KAAL,CAAV,CAAL,C;YACI,OAAO,gBAAK,Q
AAQ,CAAR,IAAL,C;;QAGf,OAAO,iB;MACX,C;KafA,C;2FAiBA,yB;MAAA,+D;MAAA,uC;QAUiB,Q;QADb,
WAAW,gB;QACE,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACI,K;UA
CJ,IAAK,WAAI,IAAJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;2FAkBA,yB;MAAA,+D;MAAA,uC;QAUiB,Q;Q
ADb,WAAW,gB;QACE,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,IAAI,CAAC,UAAU,IAAV,CAAL,C;YACI,K
;UACJ,IAAK,WAAI,IAAJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;2FAkBA,yB;MAAA,+D;MAAA,uC;QAUiB,
Q;QADb,WAAW,gB;QACE,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,IAAI,CAAC,UAAU,IAAV,CAAL,C;YA
CI,K;UACJ,IAAK,WAAI,IAAJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;2FAkBA,yB;MAAA,+D;MAAA,uC;QA
UiB,Q;QADb,WAAW,gB;QACE,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,IAAI,CAAC,UAAU,IAAV,CAAL,C
;YACI,K;UACJ,IAAK,WAAI,IAAJ,C;;QAET,OAAO,I;MACX,C;KAhBA,C;sFAkBA,yB;MAAA,kD;MAAA,4B;
QAOY,QAAR,iBAAQ,C;MACZ,C;KARA,C;sFAUA,yB;MAAA,kD;MAAA,4B;QAOY,QAAR,iBAAQ,C;MACZ,
C;KARA,C;uFAUA,yB;MAAA,kD;MAAA,4B;QAOY,QAAR,iBAAQ,C;MACZ,C;KARA,C;uFAUA,yB;MAAA,

kD;MAAA,4B;QAOY,QAAR,iBAAQ,C;MACZ,C;KARA,C;;MAgBI,IAAI,mBAAJ,C;QA Ae,OAAO,W;MACtB, WAAW,0B;MACN,WAAL,IAAK,C;MACL,OAAO,I;IACX,C;;MAQI,IAAI,mBAAJ,C;QA Ae,OAAO,W;MACtB, WAAW,0B;MACN,WAAL,IAAK,C;MACL,OAAO,I;IACX,C;;MAQI,IAAI,mBAAJ,C;QA Ae,OAAO,W;MACtB, WAAW,0B;MACN,WAAL,IAAK,C;MACL,OAAO,I;IACX,C;;MAQI,IAAI,mBAAJ,C;QA Ae,OAAO,W;MACtB, WAAW,0B;MACN,WAAL,IAAK,C;MACL,OAAO,I;IACX,C;kGAEA,yB;MAAA,8D;MAAA,uC;MAAA,4B;QA OI,OAAO,mBAaKB,cAAR,iBAAQ,CAaIB,C;MACX,C;KARA,C;kGAUA,yB;MAAA,8D;MAAA,yC;MAAA,4B; QA OI,OAAO,oBAaMB,cAAR,iBAAQ,CAAnB,C;MACX,C;KARA,C;mGAUA,yB;MAAA,8D;MAAA,yC;MAA A,4B;QA OI,OAAO,oBAaMB,cAAR,iBAAQ,CAAnB,C;MACX,C;KARA,C;mGAUA,yB;MAAA,8D;MAAA,2C; MAAA,4B;QA OI,OAAO,qBAaMB,cAAR,iBAAQ,CAApB,C;MACX,C;KARA,C;;MAgBI,IAAI,iBAAO,CAAX,C ;QACI,iB;QA hII,UAAR,iBAAQ,C;;IAmIZ,C;;MAQI,IAAI,iBAAO,CAAX,C;QACI,iB;QA III,UAAR,iBAAQ,C;;I AqIZ,C;;MAQI,IAAI,iBAAO,CAAX,C;QACI,iB;QA pII,UAAR,iBAAQ,C;;IAuIZ,C;;MAQI,IAAI,iBAAO,CAAX, C;QACI,iB;QA tII,UAAR,iBAAQ,C;;IAyIZ,C;;MAQoB,kBA+eT,cAAU,iBtBnwEO,QsBmwEjB,C;MA/eiB,mB;M AAXB,OAAiC,SnB38E1B,WmB28E0B,C;IACrC,C;;MAQoB,kBAkfT,eAAmB,UAAR,iBAAQ,CAAnB,C;MAIfiB, mB;MAAXB,OAAiC,SnBp9E1B,WmBo9E0B,C;IACrC,C;;MAQoB,kBAqfT,eAAW,iBtB/yEM,QsB+yEjB,C;MAr fiB,mB;MAAXB,OAAiC,UnB79E1B,WmB69E0B,C;IACrC,C;;MAQoB,kBAwfT,gBAAY,iBtBjzEK,QsBizEjB,C; MAxfiB,mB;MAAXB,OAAiC,UnBt+E1B,WmBs+E0B,C;IACrC,C;;MAQI,IAAI,mBAAJ,C;QA Ae,OAAO,S;MAC D,kBA0cd,cA1cA,SA0cU,QtBnwEO,QsBmwEjB,C;MA1csB,mB;MAA7B,OnBh/EO,W;ImBi/EX,C;;MAQI,IAAI, mBAAJ,C;QA Ae,OAAO,S;MACD,kBA4cd,eAAmB,UA5cnB,SA4cW,QAAQ,CAAnB,C;MA5csB,mB;MAA7B,O nB1/EO,W;ImB2/EX,C;;MAQI,IAAI,mBAAJ,C;QA Ae,OAAO,S;MACD,kBA8cd,eA9cA,SA8cW,QtB/yEM,QsB+ yEjB,C;MA9csB,mB;MAA7B,OnBpgFO,W;ImBqgFX,C;;MAQI,IAAI,mBAAJ,C;QA Ae,OAAO,S;MACD,kBAgd d,gBAhdA,SAgdY,QtBjzEK,QsBizEjB,C;MAhdsB,mB;MAA7B,OnB9gFO,W;ImB+gFX,C;;MAQI,IAAI,mBAAJ, C;QA Ae,OAAO,S;MACD,kBAkad,cAlaA,SakaU,QtBnwEO,QsBmwEjB,C;MALasB,6B;MAA7B,OnBxhFO,W;Im ByhFX,C;;MAQI,IAAI,mBAAJ,C;QA Ae,OAAO,S;MACD,kBAoad,eAAmB,UapanB,SAoaW,QAAQ,CAAnB,C; MAsB,6B;MAA7B,OnBliFO,W;ImBmiFX,C;;MAQI,IAAI,mBAAJ,C;QA Ae,OAAO,S;MACD,kBAasad,eAtaA,S AsaW,QtB/yEM,QsB+yEjB,C;MAtasB,8B;MAA7B,OnB5iFO,W;ImB6iFX,C;;MAQI,IAAI,mBAAJ,C;QA Ae,OA AO,S;MACD,kBAwad,gBAxaA,SAwaY,QtBjzEK,QsBizEjB,C;MAxasB,8B;MAA7B,OnBtjFO,W;ImBujFX,C;;M AUoB,kBAyXT,cAAU,iBtBnwEO,QsBmwEjB,C;MAzXiB,mB;MAAXB,OAAiC,YnBjkF1B,WmBikF0B,C;IACrC ,C;;MAUoB,kBA0XT,eAAmB,UAAR,iBAAQ,CAAnB,C;MA1XiB,mB;MAAXB,OAAiC,YnB5kF1B,WmB4kF0B, C;IACrC,C;;MAUoB,kBA2XT,eAAW,iBtB/yEM,QsB+yEjB,C;MA3XiB,mB;MAAXB,OAAiC,YnBvIF1B,WmBul F0B,C;IACrC,C;;MAUoB,kBA4XT,gBAAY,iBtBjzEK,QsBizEjB,C;MA5XiB,mB;MAAXB,OAAiC,YnBlmF1B,W mBkmF0B,C;IACrC,C;;MAUI,OAAO,iB;IACX,C;;MAUI,OAAO,iB;IACX,C;;MA sCI,OAAO,iB;IACX,C;;MAUI, OAAO,iB;IACX,C;8FAEA,yB;MAAA,yC;MAAA,4B;QA QI,OAAO,oBAAW,SAAX,C;MACX,C;KATA,C;4FA WA,yB;MAAA,uC;MAAA,4B;QA QI,OAAO,mBAAU,SAAV,C;MACX,C;KATA,C;8FAWA,yB;MAAA,yC;MA AA,4B;QA QI,OAAO,oBAAW,SAAX,C;MACX,C;KATA,C;gGAWA,yB;MAAA,2C;MAAA,4B;QA QI,OAAO,qB AAY,SAAZ,C;MACX,C;KATA,C;;MAkBI,OAAe,cAAR,iBAAQ,EAAC,KAAM,QAAPB,C;IACnB,C;;MASI,OA Ae,cAAR,iBAAQ,EAAC,KAAM,QAAPB,C;IACnB,C;;MASI,OAAe,cAAR,iBAAQ,EAAC,KAAM,QAAPB,C;IAC nB,C;;MASI,OAAe,cAAR,iBAAQ,EAAC,KAAM,QAAPB,C;IACnB,C;;MAQI,OAAe,gBAAR,iBAAQ,C;IACnB,C ;MAQI,OAAe,gBAAR,iBAAQ,C;IACnB,C;;MAQI,OAAe,gBAAR,iBAAQ,C;IACnB,C;;MAQI,OAAe,gBAAR,iB AAQ,C;IACnB,C;;MAUI,OAAO,0BAaA,IAAb,EAAMB,GAAnB,EAawB,GAAXB,C;IACX,C;;MAUI,OAAO,0B AAa,IAAb,EAAMB,GAAnB,EAawB,GAAXB,C;IACX,C;;MAUI,OAAO,0BAaA,IAAb,EAAMB,GAAnB,EAawB ,GAAXB,C;IACX,C;;MAUI,OAAO,0BAaA,IAAb,EAAMB,GAAnB,EAawB,GAAXB,C;IACX,C;sFAEA,yB;MtB/ 0EA,8C;MsB+0EA,kF;QAmB6D,iC;UAAA,oBAAYB,C;QAAG,0B;UAAA,aAaKB,C;QAAG,wB;UAAA,WAAgB ,c;QtB90E1H,UsB+0EA,iBtB/0EA,EsB+0EiB,WAAY,QtB/0E7B,EsB+0EsC,iBtB/0EtC,EsB+0EyD,UtB/0EzD,Es B+0EqE,QtB/0ErE,C;QsBg1EA,OAAO,W;MACX,C;KAtBA,C;wFAwBA,yB;MtB/0EA,8C;MsB+0EA,kF;QAmB +D,iC;UAAA,oBAAYB,C;QAAG,0B;UAAA,aAaKB,C;QAAG,wB;UAAA,WAAgB,c;QtB90E5H,UsB+0EA,iBtB/ 0EA,EsB+0EiB,WAAY,QtB/0E7B,EsB+0EsC,iBtB/0EtC,EsB+0EyD,UtB/0EzD,EsB+0EqE,QtB/0ErE,C;QsBg1E A,OAAO,W;MACX,C;KAtBA,C;wFAwBA,yB;MtB/6EA,8C;MsB+6EA,kF;QAmB+D,iC;UAAA,oBAAYB,C;QA AG,0B;UAAA,aAaKB,C;QAAG,wB;UAAA,WAAgB,c;QtB96E5H,UsB+6EA,iBtB/6EA,EsB+6EiB,WAAY,QtB/6

E7B,EsB+6EsC,iBtB/6EtC,EsB+6EyD,UtB/6EzD,EsB+6EqE,QtB/6ErE,C;QsBg7EA,OAAO,W;MACX,C;KAtBA,C;wFAwBA,yB;MtB/6EA,8C;MsB+6EA,kF;QAmBiE,iC;UAAA,oBAAYB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,c;QtB96E9H,UsB+6EA,iBtB/6EA,EsB+6EiB,WAAy,QtB/6E7B,EsB+6EsC,iBtB/6EtC,EsB+6EyD,UtB/6EzD,EsB+6EqE,QtB/6ErE,C;QsBg7EA,OAAO,W;MACX,C;KAtBA,C;kFAwBA,yB;MAAA,uC;MAAA,4B;QASI,OAAO,mBAAU,iBtBnwEO,QsBmwEjB,C;MACX,C;KAVA,C;oFAYYA,yB;MAAA,gD;MAAA,yC;MAAA,4B;QASI,OAAO,oBAAmB,OAAR,iBAAQ,CAAnB,C;MACX,C;KAVA,C;oFAYYA,yB;MAAA,yC;MAAA,4B;QASI,OAAO,oBAAW,iBtB/yEM,QsB+yEjB,C;MACX,C;KAVA,C;oFAYYA,yB;MAAA,2C;MAAA,4B;QASI,OAAO,qBAAy,iBtBjzEK,QsBizEjB,C;MACX,C;KAVA,C;oFAYYA,yB;MAAA,gD;MAAA,uC;MAAA,qC;QAWI,OAAO,mBAAkB,OAAR,iBAAQ,EAAO,OAAP,CAAIb,C;MACX,C;KAZA,C;oFACa,yB;MAAA,gD;MAAA,yC;MAAA,qC;QAWI,OAAO,oBAAmB,OAAR,iBAAQ,EAAO,OAAP,CAAnB,C;MACX,C;KAZA,C;oFACa,yB;MAAA,+C;MAAA,yC;MAAA,qC;QAWI,OAAO,oBAAmB,OAAR,iBAAQ,EAAO,OAAP,CAAnB,C;MACX,C;KAZA,C;oFACa,yB;MAAA,gD;MAAA,2C;MAAA,qC;QAWI,OAAO,qBAaOB,OAAR,iBAAQ,EAAO,OAAP,CAAPB,C;MACX,C;KAZA,C;4FACa,yB;MAAA,0D;MAAA,uC;MAAA,gD;QAUI,OAAO,mBAAkB,YAAR,iBAAQ,EAAy,SAAZ,EAAuB,OAAvB,CAAIb,C;MACX,C;KAXA,C;8FAaA,yB;MAAA,0D;MAAA,yC;MAAA,gD;QAUI,OAAO,oBAAmB,YAAR,iBAAQ,EAAy,SAAZ,EAAuB,OAAvB,CAAnB,C;MACX,C;KAXA,C;8FAaA,yB;MAAA,0D;MAAA,yC;MAAA,gD;QAUI,OAAO,oBAAmB,YAAR,iBAAQ,EAAy,SAAZ,EAAuB,OAAvB,CAAnB,C;MACX,C;KAXA,C;6FAaA,yB;MAAA,0D;MAAA,2C;MAAA,gD;QAUI,OAAO,qBAaOB,YAAR,iBAAQ,EAAy,SAAZ,EAAuB,OAAPB,C;MACX,C;KAXA,C;MAWByC,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,c;MACHe,OAAR,iBAAQ,EAAK,OjCv+FoB,KiCu+FzB,EAAsB,SAAtB,EAAiC,OAajC,C;IACZ,C;MAa2C,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,c;MACIE,OAAR,iBAAQ,EAAK,OF3+Fsb,KE2+F3B,EAAuB,SAAvB,EAAkC,OAaiC,C;IACZ,C;MAa2C,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,c;MACIE,OAAR,iBAAQ,EAAK,OICjiGsB,KkCiiG3B,EAAuB,SAAvB,EAAkC,OAaiC,C;IACZ,C;MAa6C,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,c;MACpE,OAAR,iBAAQ,EAAK,OhCriGwB,KgCqiG7B,EAAwB,SAAxB,EAAmC,OAAnC,C;IACZ,C;8FAQI,yB;MAAA,0D;MAAA,4B;QAAQ,OAAQ,YAAR,iBAAQ,C;MAAR,C;KAAR,C;8FAQA,yB;MAAA,0D;MAAA,4B;QAAQ,OAAQ,YAAR,iBAAQ,C;MAAR,C;KAAR,C;+FAQA,yB;MAAA,0D;MAAA,4B;QAAQ,OAAQ,YAAR,iBAAQ,C;MAAR,C;KAAR,C;+FAQA,yB;MAAA,0D;MAAA,4B;QAAQ,OAAQ,YAAR,iBAAQ,C;MAAR,C;KAAR,C;+FAQA,yB;MAAA,0D;MAAA,4B;QAAQ,OAAQ,cAAR,iBAAQ,C;MAAR,C;KAAR,C;kGAQA,yB;MAAA,8D;MAAA,4B;QAAQ,OAAQ,cAAR,iBAAQ,C;MAAR,C;KAAR,C;mGAQA,yB;MAAA,8D;MAAA,4B;QAAQ,OAAQ,cAAR,iBAAQ,C;MAAR,C;KAAR,C;mGAQA,yB;MAAA,8D;MAAA,4B;QAAQ,OAAQ,cAAR,iBAAQ,C;MAAR,C;KAAR,C;MAAaB,OAAGB,C;QAAGB,yB;QAAU,OAAO,cAAP,EAAO,sBAAP,YAAkB,OjCvoGX,K;MiCwoGjC,OA AO,cAAU,MAAV,C;IACX,C;MAUoB,UAAiB,M;MAFjC,YAAy,c;MACZ,aAAqB,UAAR,iBAAQ,EAAO,iBAAO,QAAS,KAaHB,IAAP,C;MACL,0B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QAAU,OAAO,cAAP,EAAO,sBAAP,YAAkB,OFxoGT,K;MEyoGnC,OAAO,eAAW,MAAX,C;IACX,C;MAUoB,UAAiB,M;MAFjC,YAAy,c;MACZ,aAAqB,UAAR,iBAAQ,EAAO,iBAAO,QAAS,KAaHB,IAAP,C;MACL,0B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QAAU,OAAO,cAAP,EAAO,sBAAP,YAAkB,OIC3rGT,K;MkC4rGnC,OAAO,eAAW,MAAX,C;IACX,C;MAUoB,UAAiB,M;MAFjC,YAAy,c;MACZ,aAAqB,UAAR,iBAAQ,EAAO,iBAAO,QAAS,KAaHB,IAAP,C;MACL,0B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QAAU,OAAO,cAAP,EAAO,sBAAP,YAAkB,OhC5rGP,K;MgC6rGrC,OAAO,gBAAY,MAAZ,C;IACX,C;IFAEA,yB;MAAA,uC;MtBriEA,iD;MsBqiEA,sC;QAOL,OAAO,mBtBviEA,qBsBuiEU,iBtBviEV,EsBuiEoB,QAAS,QtBviE7B,CsBuiEA,C;MACX,C;KARA,C;IFAUA,yB;MAAA,yC;MtBviEA,iD;MsBuiEA,sC;QAOL,OAAO,oBtBziEA,qBsByiEW,iBtBziEX,EsByiEqB,QAAS,QtBziE9B,CsByiEA,C;MAC

X,C;KARA,C;iFAUA,yB;MAAA,yC;MtBzkEA,iD;MsBykEA,sC;QAOI,OAAO,oBtB3kEA,qBsB2kEW,iBtB3kEX
,EsB2kEqB,QAAS,QtB3kE9B,CsB2kEA,C;MACX,C;KARA,C;iFAUA,yB;MAAA,2C;MtB3kEA,iD;MsB2kEA,s
C;QAOI,OAAO,qBtB7kEA,qBsB6kEY,iBtB7kEZ,EsB6kEsB,QAAS,QtB7kE/B,CsB6kEA,C;MACX,C;KARA,C;;
MAKBI,IAAI,iBAAO,CAAX,C;QAAC,YAAU,SAAV,C;IACIB,C;;MAUI,IAAI,iBAAO,CAAX,C;QAAC,YAAU,S
AAV,C;IACIB,C;;MAUI,IAAI,iBAAO,CAAX,C;QAAC,YAAU,SAAV,C;IACIB,C;;MAUI,IAAI,iBAAO,CAAX,C
;QAAC,YAAU,SAAV,C;IACIB,C;;MAUI,OAAO,iBtB1tFiB,Q;IsB2tF5B,C;;MAUI,OAAO,iBtBjtFiB,Q;IsBktF5B,
C;8FAEA,yB;MAAA,gD;MAAA,4B;QAQI,OAAe,OAAR,iBAAQ,C;MACnB,C;KATA,C;;MAmBI,OAAO,iBtBjv
FiB,Q;IsBkvF5B,C;;MAQuB,wB;QAAW,qCAAK,KAAL,C;MAAX,C;IAAA,C;;MAAnB,OAAO,iBAAM,cAAN,E
AAY,8BAAZ,C;IACX,C;;MAQuB,wB;QAAW,qCAAK,KAAL,C;MAAX,C;IAAA,C;;MAAnB,OAAO,iBAAM,c
AAN,EAAY,gCAAZ,C;IACX,C;;MAQuB,wB;QAAW,qCAAK,KAAL,C;MAAX,C;IAAA,C;;MAAnB,OAAO,iB
AAM,cAAN,EAAY,gCAAZ,C;IACX,C;;MAQuB,wB;QAAW,qCAAK,KAAL,C;MAAX,C;IAAA,C;;MAAnB,OA
AO,iBAAM,cAAN,EAAY,gCAAZ,C;IACX,C;;MAQ4B,wB;QAAW,yBAAK,KAAL,C;MAAX,C;IAAA,C;;MAA
xB,OLn9GO,eAAW,+BKm9GA,gBLn9GA,GAAgB,kBKm9GV,8BLn9GU,CAAhB,CAAX,C;IKo9GX,C;gGAEA,
yB;MAAA,yC;MAAA,4B;QAQI,OAAO,oBAAW,StBnzFM,QsBmzFjB,C;MACX,C;KATA,C;;MAiB2B,wB;QAA
W,wBAAK,KAAL,C;MAAX,C;IAAA,C;;MAAvB,OJv+GO,cAAU,gClu+GA,gBJv+GA,GAAe,iBtu+GT,6BJv+G
S,CAAf,CAAV,C;Iiw+GX,C;8FAEA,yB;MAAA,uC;MAAA,4B;QAQI,OAAO,mBAAU,StBnzFO,QsBmzFjB,C;M
ACX,C;KATA,C;;MAiB4B,wB;QAAW,yBAAK,KAAL,C;MAAX,C;IAAA,C;;MAAxB,OH3/GO,eAAW,kBG2/G
A,gBH3/GA,EAAGB,kBG2/GV,8BH3/GU,CAAhB,CAAX,C;IG4/GX,C;gGAEA,yB;MAAA,gD;MAAA,yC;MAA
A,4B;QAQI,OAAO,oBAAGB,OAAL,SAAK,CAAhB,C;MACX,C;KATA,C;;MAiB6B,wB;QAAW,0BAAK,KAAL
,C;MAAX,C;IAAA,C;;MAAzB,OD/gHO,gBAAY,gCC+gHA,gBD/gHA,GAAiB,mBC+gHX,+BD/gHW,CAAjB,C
AAZ,C;ICghHX,C;kGAEA,yB;MAAA,2C;MAAA,4B;QAQI,OAAO,qBAAY,StBr2FK,QsBq2FjB,C;MACX,C;K
ATA,C;uFAWA,yB;MAAA,+D;MAwCA,gD;MAxCA,uC;QAOW,kBAAU,gB;QAwCD,Q;QAAA,2B;QAAhB,OA
AgB,cAAhB,C;UAAgB,yB;UACZ,WazC6B,SAyCIB,CAAU,OAAV,C;UACC,OAAZ,WAAy,EAAO,IAAP,C;;Q
A1ChB,OA4CO,W;MA3CX,C;KARA,C;iFAUA,yB;MAAA,+D;MA4CA,gD;MA5CA,uC;QAOW,kBAAU,gB;Q
A4CD,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,WA7C6B,SA6CIB,CAAU,OAAV,C;UACC,O
AAZ,WAAy,EAAO,IAAP,C;;QA9ChB,OAgDO,W;MA/CX,C;KARA,C;iFAUA,yB;MAAA,+D;MAgDA,gD;MA
hDA,uC;QAOW,kBAAU,gB;QAgDD,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,WajD6B,SAi
DIB,CAAU,OAAV,C;UACC,OAAZ,WAAy,EAAO,IAAP,C;;QAIDhB,OAoDO,W;MAnDX,C;KARA,C;uFAUA,
yB;MAAA,+D;MAoDA,gD;MApDA,uC;QAOW,kBAAU,gB;QAoDD,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;U
AAgB,yB;UACZ,WArD6B,SAqDIB,CAAU,OAAV,C;UACC,OAAZ,WAAy,EAAO,IAAP,C;;QAtDhB,OAwDO,
W;MAvDX,C;KARA,C;2FAUA,yB;MAAA,gD;MAAA,oD;QAObB,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UA
AgB,yB;UACZ,WAAW,UAAU,OAAV,C;UACC,OAAZ,WAAy,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;K
AZA,C;2FAcA,yB;MAAA,gD;MAAA,oD;QAObB,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,
WAAW,UAAU,OAAV,C;UACC,OAAZ,WAAy,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KAZA,C;2FAcA,y
B;MAAA,gD;MAAA,oD;QAObB,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,WAAW,UAAU,O
AAV,C;UACC,OAAZ,WAAy,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KAZA,C;2FAcA,yB;MAAA,gD;MA
AA,oD;QAObB,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,WAAW,UAAU,OAAV,C;UACC,O
AAZ,WAAy,EAAO,IAAP,C;;QAEhB,OAAO,W;MACX,C;KAZA,C;uFAcA,yB;MAAA,wE;MA4HA,+D;MA5H
A,yC;QAYW,kBAAU,oB;QA4HD,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,UA7HoD,WA6H
1C,CAAY,OAAZ,C;U7BhiHP,U;UADP,Y6BkiHe,W7BliHH,W6BkiHwB,G7BliHxB,C;UACL,IAAI,aAAJ,C;YAC
H,a6BgiHuC,gB;YAA5B,W7B/hHX,a6B+hHgC,G7B/hHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6B4hHA,iB;U
ACA,IAAK,WAAI,OAAJ,C;;QA/HT,OAIIO,W;MAhIX,C;KAbA,C;uFAeA,yB;MAAA,wE;MAiIA,+D;MAjIA,yC
;QAYW,kBAAU,oB;QAIID,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,UAlIqD,WakI3C,CAA
Y,OAAZ,C;U7BpjHP,U;UADP,Y6BsjHe,W7BtjHH,W6BsjHwB,G7BtjHxB,C;UACL,IAAI,aAAJ,C;YACH,a6Boj
HuC,gB;YAA5B,W7BnjHX,a6BmjHgC,G7BnjHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6BgjHA,iB;UACA,IA
AK,WAAI,OAAJ,C;;QApIT,OAsIO,W;MARIX,C;KAbA,C;uFAeA,yB;MAAA,wE;MAsIA,+D;MAtIA,yC;QAYW,
kBAAU,oB;QAsID,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,UAvIqD,WauI3C,CAAY,OAAZ
,C;U7BxkHP,U;UADP,Y6B0kHe,W7B1kHH,W6B0kHwB,G7B1kHxB,C;UACL,IAAI,aAAJ,C;YACH,a6BwkHuC

,gB;YAA5B,W7BvkHX,a6BukHgC,G7BvkHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6BokHA,iB;UACA,IAAK,WAAL,OAAJ,C;;QAZIT,OA2IO,W;MA1IX,C;KAbA,C;uFAeA,yB;MAAA,wE;MA2IA,+D;MA3IA,yC;QAYW,kBAAU,oB;QA2ID,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA5IsD,WA4I5C,CAAY,OAAZ,C;U7B5IHP,U;UADP,Y6B8IHe,W7B9IHH,W6B8IHwB,G7B9IHxB,C;UACL,IAAI,aAAJ,C;YACH,a6B4IHuC,gB;YAA5B,W7B3IHX,a6B2IHgC,G7B3IHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6BwiHA,iB;UACA,IAAK,WAAL,OAAJ,C;;QA9IT,OAgJO,W;MA/IX,C;KAbA,C;uFAeA,yB;MAAA,wE;MAgJA,+D;MAhJA,yD;QAaW,kBAAU,oB;QAqJD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAjJiD,WaiJvC,CAAY,OAAZ,C;U7BjnHP,U;UADP,Y6BmnHe,W7BnnHH,W6BmnHwB,G7BnnHxB,C;UACL,IAAI,aAAJ,C;YACH,a6BinHuC,gB;YAA5B,W7BhnHX,a6BgnHgC,G7BhnHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6B6mHA,iB;UACA,IAAK,WAnJyD,cAmJrD,CAAE,OAAf,CAAJ,C;;QAnJT,OAgJO,W;MApJX,C;KAdA,C;uFagBA,yB;MAAA,wE;MAqJA,+D;MArJA,yD;QAaW,kBAAU,oB;QAqJD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAtJiD,WAsJvC,CAAY,OAAZ,C;U7BtoHP,U;UADP,Y6BwoHe,W7BxoHH,W6BwoHwB,G7BxoHxB,C;UACL,IAAI,aAAJ,C;YACH,a6BsoHuC,gB;YAA5B,W7BroHX,a6BqoHgC,G7BroHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6BkoHA,iB;UACA,IAAK,WaxJyD,cAwJrD,CAAE,OAAf,CAAJ,C;;QAxJT,OA0JO,W;MAzJX,C;KAdA,C;uFagBA,yB;MAAA,wE;MA0JA,+D;MA1JA,yD;QAaW,kBAAU,oB;QA0JD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UA3JiD,WA2JvC,CAAY,OAAZ,C;U7B3pHP,U;UADP,Y6B6pHe,W7B7pHH,W6B6pHwB,G7B7pHxB,C;UACL,IAAI,aAAJ,C;YACH,a6B2pHuC,gB;YAA5B,W7B1pHX,a6B0pHgC,G7B1pHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6BupHA,iB;UACA,IAAK,WA7JyD,cA6JrD,CAAE,OAAf,CAAJ,C;;QA7JT,OA+JO,W;MA9JX,C;KAdA,C;uFagBA,yB;MAAA,wE;MA+JA,+D;MA/JA,yD;QAaW,kBAAU,oB;QA+JD,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAhKiD,WagKvC,CAAY,OAAZ,C;U7BhrHP,U;UADP,Y6BkrHe,W7BlrHH,W6BkrHwB,G7BlrHxB,C;UACL,IAAI,aAAJ,C;YACH,a6BgrHuC,gB;YAA5B,W7B/qHX,a6B+qHgC,G7B/qHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6B4qHA,iB;UACA,IAAK,WAlKyD,cAkKrD,CAAE,OAAf,CAAJ,C;;QA1KT,OAoKO,W;MAAnKX,C;KAdA,C;2FagBA,yB;MAAA,+D;MAAA,sD;QAYoB,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAAY,OAAZ,C;U7BhiHP,U;UADP,Y6BkiHe,W7BliHH,W6BkiHwB,G7BliHxB,C;UACL,IAAI,aAAJ,C;YACH,a6BgiHuC,gB;YAA5B,W7B/hHX,a6B+hHgC,G7B/hHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6B4hHA,iB;UACA,IAAK,WAAL,OAAJ,C;;QAET,OAAO,W;MACX,C;KAIBA,C;2FAoBA,yB;MAAA,+D;MAAA,sD;QAYoB,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAAY,OAAZ,C;U7BpjHP,U;UADP,Y6BsjHe,W7BtjHH,W6BsjHwB,G7BtjHxB,C;UACL,IAAI,aAAJ,C;YACH,a6BojHuC,gB;YAA5B,W7BnjHX,a6BmjHgC,G7BnjHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6BgjHA,iB;UACA,IAAK,WAAL,OAAJ,C;;QAET,OAAO,W;MACX,C;KAIBA,C;2FAoBA,yB;MAAA,+D;MAAA,sD;QAYoB,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAAY,OAAZ,C;U7BxkHP,U;UADP,Y6B0kHe,W7B1kHH,W6B0kHwB,G7B1kHxB,C;UACL,IAAI,aAAJ,C;YACH,a6BwkHuC,gB;YAA5B,W7BvkHX,a6BukHgC,G7BvkHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6BokHA,iB;UACA,IAAK,WAAL,OAAJ,C;;QAET,OAAO,W;MACX,C;KAIBA,C;2FAoBA,yB;MAAA,+D;MAAA,sD;QAYoB,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAAY,OAAZ,C;U7B5IHP,U;UADP,Y6B8IHe,W7B9IHH,W6B8IHwB,G7B9IHxB,C;UACL,IAAI,aAAJ,C;YACH,a6B4IHuC,gB;YAA5B,W7B3IHX,a6B2IHgC,G7B3IHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6BwiHA,iB;UACA,IAAK,WAAL,OAAJ,C;;QAET,OAAO,W;MACX,C;KAIBA,C;2FAoBA,yB;MAAA,+D;MAAA,sE;QAaoB,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAAY,OAAZ,C;U7BjnHP,U;UADP,Y6BmnHe,W7BnnHH,W6BmnHwB,G7BnnHxB,C;UACL,IAAI,aAAJ,C;YACH,a6BinHuC,gB;YAA5B,W7BhnHX,a6BgnHgC,G7BhnHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6B6mHA,iB;UACA,IAAK,WAAL,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAnBA,C;2FAqBA,yB;MAAA,+D;MAAA,sE;QAaoB,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAAY,OAAZ,C;U7BtoHP,U;UADP,Y6BwoHe,W7BxoHH,W6BwoHwB,G7BxoHxB,C;UACL,IAAI,aAAJ,C;YACH,a6BsoHuC,gB;YAA5B,W7BroHX,a6BqoHgC,G7BroHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6BkoHA,iB;UACA,IAAK,WAAL,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAnBA,C;2FAqBA,yB;MAAA,+D;MAAA,sE;QAaoB,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAAY,OAAZ,C;U7B3pHP,U;UADP,Y6B6pHe,W7B7pHH,W6B6pHwB,G7B7pHxB,C;UACL,IAAI,aAAJ,C;YACH,a6B2pHuC,gB;YAA5B,W7B1pHX,a6B0pHgC,G7B1pHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6BupHA,iB;UACA,IAAK,WAAL,eAAe,OAAf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAnBA,C;2FAqB

A,yB;MAAA,+D;MAAA,sE;QAaoB,Q;QAAA,2B;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UACZ,UAAU,YAA Y,
OAAZ,C;U7BhrHP,U;UADP,Y6BkrHe,W7BlrHH,W6BkrHwB,G7BlrHxB,C;UACL,IAAI,aAAJ,C;YACH,a6BgrH
uC,gB;YAA5B,W7B/qHX,a6B+qHgC,G7B/qHhC,EAAS,MAAT,C;YACA,e;;;YAEA,c;;U6B4qHA,iB;UACA,IAA
K,WAAI,eAAe,OAaf,CAAJ,C;;QAET,OAAO,W;MACX,C;KAnBA,C;+EAqBA,yB;MAAA,gE;MAAA,uC;QAU
W,kBAAM,eAAa,cAAb,C;QAsKA,Q;QAAA,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAA Y,WAvKiB,SAuK
b,CAAU,IAAV,CAAJ,C;;QAvKhB,OAwKO,W;MAvKX,C;KAXA,C;+EAaA,yB;MAAA,gE;MAAA,uC;QAUW,k
BAAM,eAAa,cAAb,C;QAsKA,Q;QAAA,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAA Y,WAvKiB,SAuKb,C
AAU,IAAV,CAAJ,C;;QAvKhB,OAwKO,W;MAvKX,C;KAXA,C;8EAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBA
AM,eAAa,cAAb,C;QAsKA,Q;QAAA,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAA Y,WAvKiB,SAuKb,CAA
U,IAAV,CAAJ,C;;QAvKhB,OAwKO,W;MAvKX,C;KAXA,C;+EAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAAM
,eAAa,cAAb,C;QAsKA,Q;QAAA,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAA Y,WAvKiB,SAuKb,CAAU,I
AAV,CAAJ,C;;QAvKhB,OAwKO,W;MAvKX,C;KAXA,C;4FAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAaA,eA
Aa,cAAb,C;QAqDP,gB;QADb,YAA Y,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAA Y,WAtDwB,S
AsDpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QAtDhB,OAuDO,W;MatDX,C;KAXA,C;6FAa
A,yB;MAAA,gE;MAAA,uC;QAUW,kBAaA,eAAa,cAAb,C;QAwDP,gB;QADb,YAA Y,C;QACC,2B;QAAb,OAA
a,cAAb,C;UAAa,sB;UACT,WAA Y,WAZDwB,SAyDpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;
;QAZDhB,OA0DO,W;MAZDX,C;KAXA,C;6FAaA,yB;MAAA,gE;MAAA,uC;QAUW,kBAaA,eAAa,cAAb,C;QA2
DP,gB;QADb,YAA Y,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,WAA Y,WA5DwB,SA4DpB,EAAU,c
AAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA5DhB,OA6DO,W;MA5DX,C;KAXA,C;4FAaA,yB;MAAA,g
E;MAAA,uC;QAUW,kBAaA,eAAa,cAAb,C;QA8DP,gB;QADb,YAA Y,C;QACC,2B;QAAb,OAAa,cAAb,C;UAA
a,sB;UACT,WAA Y,WA/DwB,SA+DpB,EAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;QA/DhB,OA gE
O,W;MA/DX,C;KAXA,C;;MAwBiB,UACiB,M;MAF9B,YAA Y,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;Q
ACT,WAA Y,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAa
iB,UACiB,M;MAF9B,YAA Y,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,WAA Y,WAAI,WAAU,cAA
V,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAaiB,UACiB,M;MAF9B,YAA Y,C;
MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,WAA Y,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,IAAn
B,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAaiB,UACiB,M;MAF9B,YAA Y,C;MACC,2B;MAAb,OAAa,cAAb,C;
QAAa,sB;QACT,WAA Y,WAAI,WAAU,cAAV,EAAU,sBAAV,WAAmB,IAAnB,CAAJ,C;;MACHB,OAAO,W;IA
CX,C;;MAUiB,Q;MAAA,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,WAA Y,WAAI,UAAU,IAAV,CAAJ,C;;MA
ChB,OAAO,W;IACX,C;;MAUiB,Q;MAAA,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,WAA Y,WAAI,UAAU,I
AAV,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAUiB,Q;MAAA,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,WAA
Y,WAAI,UAAU,IAAV,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAUiB,Q;MAAA,2B;MAAb,OAAa,cAAb,C;QAA
a,sB;QACT,WAA Y,WAAI,UAAU,IAAV,CAAJ,C;;MACHB,OAAO,W;IACX,C;;MAS4B,mB;QAAE,gC;MAAF,C
;IAAA,C;;MAAxB,OAAO,qBAAiB,8BAAjB,C;IACX,C;;MAS4B,mB;QAAE,gC;MAAF,C;IAAA,C;;MAAxB,OA
AO,qBAAiB,8BAAjB,C;IACX,C;;MAS4B,mB;QAAE,gC;MAAF,C;IAAA,C;;MAAxB,OAAO,qBAAiB,8BAAjB,
C;IACX,C;;MAS4B,mB;QAAE,gC;MAAF,C;IAAA,C;;MAAxB,OAAO,qBAAiB,8BAAjB,C;IACX,C;;MAWOB,Q
;MAAA,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QAAM,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,OAA
O,K;;MACtD,OAAO,I;IACX,C;;MAWOB,Q;MAAA,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QAAM,IAAI,C
AAC,UAAU,OAAV,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I;IACX,C;;MAWOB,Q;MAAA,2B;MAAhB,OA
AGB,cAAhB,C;QAAGB,yB;QAAM,IAAI,CAAC,UAAU,OAAV,CAAL,C;UAAyB,OAAO,K;;MACtD,OAAO,I;IA
CX,C;;MAWOB,Q;MAAA,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QAAM,IAAI,CAAC,UAAU,OAAV,CAA
L,C;UAAyB,OAAO,K;;MACtD,OAAO,I;IACX,C;;+EAEA,yB;MAAA,0C;MAAA,4B;QASI,OAAe,IAAR,iBAAQ,
C;MACnB,C;KAVA,C;+EAYA,yB;MAAA,0C;MAAA,4B;QASI,OAAe,IAAR,iBAAQ,C;MACnB,C;KAVA,C;+E
AYA,yB;MAAA,0C;MAAA,4B;QASI,OAAe,IAAR,iBAAQ,C;MACnB,C;KAVA,C;+EAYA,yB;MAAA,0C;MAA
A,4B;QASI,OAAe,IAAR,iBAAQ,C;MACnB,C;KAVA,C;;MAqBoB,Q;MAAA,2B;MAAhB,OAAGB,cAAhB,C;Q
AAGB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,I;;MACrD,OAAO,K;IACX,C;;MAWOB,Q;MAA
A,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,I;;MACrD,
OAAO,K;IACX,C;;MAWOB,Q;MAAA,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QAAM,IAAI,UAAU,OAAV,

CAAJ,C;UAAwB,OAAO,I;MACrD,OAAO,K;IACX,C;;MAWOB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,I;MACrD,OAAO,K;IACX,C;;MAUOB,Q;MADhB,YAAY,C;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,qB;;MAC9C,OAAO,K;IACX,C;;MAUOB,Q;MADhB,YAAY,C;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAm,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,qB;;MAC9C,OAAO,K;IACX,C;;MAUOB,Q;MADhB,YAAY,C;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,qB;;MAC9C,OAAO,K;IACX,C;;MAUOB,Q;MADhB,YAAY,C;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,qB;;MAC9C,OAAO,K;IACX,C;;MAUOB,Q;MADhB,kBAAkB,O;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;IACX,C;;MAUOB,Q;MADhB,kBAAkB,O;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;IACX,C;;MAUOB,Q;MADhB,kBAAkB,O;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;IACX,C;;MAUOB,Q;MADhB,kBAAkB,O;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;IACX,C;;MAUOB,Q;MADhB,kBAAkB,O;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,UAAU,WAAV,EAAuB,OAAvB,C;;MACpC,OAAO,W;IACX,C;;MAcoB,UAA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAAnB,EAAgC,OAAhC,C;;MACpC,OAAO,W;IACX,C;;MAcoB,UAA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAAnB,EAAgC,OAAhC,C;;MACpC,OAAO,W;IACX,C;;MAcoB,UAA8B,M;MAF9C,YAAY,C;MACZ,kBAAkB,O;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,cAAc,WAAU,cAAV,EAAU,sBAAV,WAAmB,WAAAnB,EAAgC,OAAhC,C;;MACpC,OAAO,W;IACX,C;0FAEA,yB;MAlyCI,8D;MAkyCJ,gD;QAUOC,Q;QAHhC,YAzyCgB,cAAR,iBAAQ,C;QA0yChB,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAbA,C;2FAeA,yB;MAzyCI,8D;MAyyCJ,gD;QAUOC,Q;QAHhC,YAhzCgB,cAAR,iBAAQ,C;QAizChB,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAbA,C;2FAeA,yB;MAhzCI,8D;MAGzCJ,gD;QAUOC,Q;QAHhC,YAvzCgB,cAAR,iBAAQ,C;QAWzChB,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAbA,C;2FAeA,yB;MAvzCI,8D;MAuzCJ,gD;QAUOC,Q;QAHhC,YA9zCgB,cAAR,iBAAQ,C;QA+zChB,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,YAAJ,EAAI,oBAAJ,QAAV,EAAwB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAbA,C;yGAeA,yB;MA9ICI,8D;MA8ICJ,gD;QAUUI,YAx2CgB,cAAR,iBAAQ,C;QAy2ChB,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAHV,EAAiB,sBAAI,KAHJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAjBA,C;yGAmBA,yB;MAz2CI,8D;MAy2CJ,gD;QAUUI,YAn3CgB,cAAR,iBAAQ,C;QAo3ChB,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,sBAAI,KAHJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAjBA,C;yGAmBA,yB;MAp3CI,8D;MAo3CJ,gD;QAUUI,YA93CgB,cAAR,iBAAQ,C;QA+3ChB,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,sBAAI,KAHJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAjBA,C;yGAmBA,yB;MA/3CI,8D;MA+3CJ,gD;QAUUI,YAz4CgB,cAAR,iBAAQ,C;QA04ChB,kBAAkB,O;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,sBAAI,KAHJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAjBA,C;;MA0BoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,OAAO,OAAP,C;;IAC1B,C;;MASoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,OAAO,OAAP,C;;IAC1B,C;;MASoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAM,OAAO,OAAP,C;;IAC1B,C;;MASyB,UAAa,M;MAD1B,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QAAM,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;IACvB,C;;MAYiB,UAAa,M;MAD1B,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QAAM,QAAO,cAAP,EAAO,sBAAP,WAAgB,IAAhB,C;;IACvB,C;;MAYiB,UAAa,M;MAD1B,YAAY,C;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,sB;QAAM,QAAO,cAAP,EAA

O,sBAAP,WAAgB,IAAhB,C;;IACvB,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACTb,UAAU,sBA
AK,CAAL,C;MACG,OA9gDG,gBAAR,iBAAQ,C;MA8gDhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;Q
ACR,IjCrtJ8D,YiCqtJ1D,GjCrtJ2E,KAAjB,EiCqtJpD,CjCrtJiF,KAA7B,CiCqtJ1D,IAAJ,C;UAAa,MAAM,C;;MAE
vB,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACTb,UAAU,sBAAK,CAAL,C;MAC
G,OArhDG,gBAAR,iBAAQ,C;MAqhDhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IF5tJ+D,aE4tJ
3D,GF5tJ6E,KAAIB,EE4tJrD,CF5tJmF,KAA9B,CE4tJ3D,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;
MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACTb,UAAU,sBAAK,CAAL,C;MACG,OA5hDG,gBAAR,iB
AAQ,C;MA4hDhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IICnwJ4E,0BkCmwJxE,GlCnnJ8B,K
AAL,GAAiB,GAhJ8B,EkCmwJIE,CiCnnJwB,KAAL,GAAiB,GAhJ8B,CkCmwJxE,IAAJ,C;UAAa,MAAM,C;;MA
EvB,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACTb,UAAU,sBAAK,CAAL,C;MA
CG,OAniDG,gBAAR,iBAAQ,C;MAmiDhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IhC1wJ6E,0
BgC0wJzE,GhCjoJ8B,KAAL,GAAiB,KAZI+B,EgC0wJnE,ChCjoJwB,KAAL,GAAiB,KAZI+B,CgC0wJzE,IAAJ,C
;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;mFAEA,yB;MAIkDI,8D;MAkkDJ,sC;QASI,IAAI,mBAAJ,C;UAA
e,OAAO,I;QACtB,cAAc,sBAAK,CAAL,C;QACd,gBA7kDgB,cA6kDA,SA7kDR,QAAQ,C;QA8kDhB,IAAI,cAAa
,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,s
BAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,W
AAW,C;;;QAGnB,OAAO,O;MACX,C;KAvBA,C;mFAyBA,yB;MAInDI,8D;MAmIDJ,sC;QASI,IAAI,mBAAJ,C;
UAAe,OAAO,I;QACtB,cAAc,sBAAK,CAAL,C;QACd,gBA9IDgB,cA8IDA,SA9IDR,QAAQ,C;QA+IDhB,IAAI,c
AAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QA
AQ,sBAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YA
CV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KAvBA,C;mFAyBA,yB;MApmDI,8D;MAomDJ,sC;QASI,IAAI,mB
AAJ,C;UAAe,OAAO,I;QACtB,cAAc,sBAAK,CAAL,C;QACd,gBA/mDgB,cA+mDA,SA/mDR,QAAQ,C;QAGnDh
B,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;
UACI,QAAQ,sBAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UA
AU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KAvBA,C;mFAyBA,yB;MArnDI,8D;MAqnDJ,sC;QASI,I
AAI,mBAAJ,C;UAAe,OAAO,I;QACtB,cAAc,sBAAK,CAAL,C;QACd,gBAhoDgB,cAgoDA,SAhoDR,QAAQ,C;
QAioDhB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,S
AAb,M;UACI,QAAQ,sBAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;Y
ACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KAvBA,C;;MAiCiB,Q;MAFb,IAAI,mBAAJ,C;Q
AAe,OAAO,I;MACTb,UAAU,sBAAK,CAAL,C;MACG,OA9qDG,gBAAR,iBAAQ,C;MA8qDhB,aAAU,CAAV,iB
;QACI,QAAQ,sBAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UA
AoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACTb,UAAU,s
BAAK,CAAL,C;MACG,OArrDG,gBAAR,iBAAQ,C;MAqrDhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C
;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,
G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACTb,UAAU,sBAAK,CAAL,C;MACG,OA5rDG
,gBAAR,iBAAQ,C;MA4rDhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GA
AR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IA
AI,mBAAJ,C;QAAe,OAAO,I;MACTb,UAAU,sBAAK,CAAL,C;MACG,OAnsDG,gBAAR,iBAAQ,C;MAmsDhB,
aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6
B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;
MACTb,UAAU,sBAAK,CAAL,C;MACG,OA1uDG,gBAAR,iBAAQ,C;MA0uDhB,aAAU,CAAV,iB;QACI,QAAQ
,sBAAK,CAAL,C;QACR,IjCj7J8D,YiCi7J1D,GjCj7J2E,KAAjB,EiCi7JpD,CjCj7JiF,KAA7B,CiCi7J1D,IAAJ,C;U
AAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACTb,UAAU,s
BAAK,CAAL,C;MACG,OAjvDG,gBAAR,iBAAQ,C;MAivDhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C
;QACR,IFx7J+D,aEw7J3D,GFx7J6E,KAAIB,EEw7JrD,CFx7JmF,KAA9B,CEw7J3D,IAAJ,C;UAAa,MAAM,C;;M
AEvB,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACTb,UAAU,sBAAK,CAAL,C;M
ACG,OAxvDG,gBAAR,iBAAQ,C;MAwvDhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IIC/9J4E,
0BkC+9JxE,GIC/0J8B,KAAL,GAAiB,GAhJ8B,EkC+9JIE,CiC/0JwB,KAAL,GAAiB,GAhJ8B,CkC+9JxE,IAAJ,C;

UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA/vDG,gBAAR,iBAAQ,C;MA+vDhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IhCt+J6E,0BgCs+JzE,GhC71J8B,KAAL,GAAiB,KAZI+B,EgCs+JnE,ChC71JwB,KAAL,GAAiB,KAZI+B,CgCs+JzE,IAAJ,C;UAAa,MAAM,C;;MAEvB,OAAO,G;IACX,C;mFAEA,yB;MA9xDI,8D;MA8xDJ,sC;QASI,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,cAAc,sBAAK,CAAL,C;QACd,gBAzyDgB,cAyyDA,SAzyDR,QAAQ,C;QA0yDhB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,sBAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KAvBA,C;mFAyBA,yB;MA/yDI,8D;MA+yDJ,sC;QASI,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,cAAc,sBAAK,CAAL,C;QACd,gBA1zDgB,cA0zDA,SA1zDR,QA AQ,C;QA2zDhB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,sBAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KAvBA,C;mFAyBA,yB;MAh0DI,8D;MAG0 DJ,sC;QASI,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,cAAc,sBAAK,CAAL,C;QACd,gBA30DgB,cA20DA,SA30DR,QAAQ,C;QA40DhB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,sBAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KAvBA,C;mFAyBA,yB;MAj1DI,8D;MAi1DJ,sC;QASI,IAAI,mBAAJ,C;UAAe,OAAO,I;QACtB,cAAc,sBAAK,CAAL,C;QACd,gBA51DgB,cA41DA,SA51DR,QAAQ,C;QA61DhB,IAAI,cAAa,CAAjB,C;UAAoB,OAAO,O;QAC3B,eAAe,SAAS,OAAT,C;QACf,aAAU,CAAV,OAAa,SAAb,M;UACI,QAAQ,sBAAK,CAAL,C;UACR,QAAQ,SAAS,CAAT,C;UACR,IAAI,2BAAW,CAAX,KAAJ,C;YACI,UAAU,C;YACV,WAAW,C;;;QAGnB,OAAO,O;MACX,C;KAvBA,C;;MAiCiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA14DG,gBAAR,iBAAQ,C;MA04Dh B,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAj5DG,gBAAR,iBAAQ,C;MAi5DhB,aAAU,CAAV,iB;QACI,QAA Q,sBAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OAx5DG,gBAAR,iBAAQ,C;MAw5DhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAUiB,Q;MAFb,IAAI,mBAAJ,C;QAAe,OAAO,I;MACtB,UAAU,sBAAK,CAAL,C;MACG,OA/5DG,gBAAR,iBAA Q,C;MA+5DhB,aAAU,CAAV,iB;QACI,QAAQ,sBAAK,CAAL,C;QACR,IAAI,UAAW,SAAQ,GAAR,EAAa,CAAb,CAAX,GAA6B,CAAjC,C;UAAoC,MAAM,C;;MAE9C,OAAO,G;IACX,C;;MAWI,OAAO,mB;IACX,C;;MAWI,OAAO,mB;IACX,C;;MAWI,OAAO,mB;IACX,C;;MAWoB,Q;MAAA,2B;MAAh B,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,K;;MACrD,OAAO,I;IAC X,C;;MAWoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAA wB,OAAO,K;;MACrD,OAAO,I;IACX,C;;MAWoB,Q;MAAA,2B;MAAhB,OAAgB,cAAhB,C;QAAGB,yB;QAAM ,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,K;;MACrD,OAAO,I;IACX,C;;MAWoB,Q;MAAA,2B;MAAhB,O AAGB,cAAhB,C;QAAGB,yB;QAAM,IAAI,UAAU,OAAV,CAAJ,C;UAAwB,OAAO,K;;MACrD,OAAO,I;IACX,C ;qFAEA,yB;MAAA,4F;MALiEI,8D;MAKiEJ,uC;QAUqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+B AA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OA5iED,cAAR,iBAAQ,C;QA4iEhB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAuB,sBAAK,KAAL,CAAvB,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;qFAgBA,yB;MA AA,4F;MA1iEI,8D;MA0iEJ,uC;QAUqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QACD,OAjED,cAAR,iBAAQ,C;QAojEhB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAuB,sBAAK,KAAL,CAAvB,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;qFAgBA,yB;MAAA,4F;MALjEI,8 D;MAkjEJ,uC;QAUqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK ,CAAL,C;QACD,OA5jED,cAAR,iBAAQ,C;QA4jEhB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAuB,sB AAK,KAAL,CAAvB,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;qFAgBA,yB;MAAA,4F;MA1jEI,8D;MA0jEJ,uC;Q AUqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAAkB,sBAAK,CAAL,C;QAC D,OApkED,cAAR,iBAAQ,C;QAokEhB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,WAAV,EAAuB,sBAAK,KAAL,C

AAvB,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;mGAgBA,yB;MAAA,4F;MAImEI,8D;MAkmEJ,uC;QAaqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAakB,sBAAK,CAAL,C;QACD,OA/mED,cAAR,iBAAQ,C;QA+mEhB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,sBAAK,KAAL,CAA9B,C;;QAEIB,OAAO,W;MACX,C;KAjBA,C;mGAmBA,yB;MAAA,4F;MA7mEI,8D;MA6mEJ,uC;QAaqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAakB,sBAAK,CAAL,C;QACD,OAInED,cAAR,iBAAQ,C;QA0nEhB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,sBAAK,KAAL,CAA9B,C;;QAEIB,OAAO,W;MACX,C;KAjBA,C;mGAmBA,yB;MAAA,4F;MAxnEI,8D;MAwnEJ,uC;QAaqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAakB,sBAAK,CAAL,C;QACD,OArOED,cAAR,iBAAQ,C;QAqoEhB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,sBAAK,KAAL,CAA9B,C;;QAEIB,OAAO,W;MACX,C;KAjBA,C;mGAmBA,yB;MAAA,4F;MAnoEI,8D;MAmoEJ,uC;QAaqB,Q;QAHjB,IAAI,mBAAJ,C;UACI,MAAM,mCAA8B,+BAA9B,C;QACV,kBAakB,sBAAK,CAAL,C;QACD,OAhpED,cAAR,iBAAQ,C;QAgpEhB,iBAAc,CAAd,yB;UACI,cAAc,UAAU,KAAV,EAAiB,WAAjB,EAA8B,sBAAK,KAAL,CAA9B,C;;QAEIB,OAAO,W;MACX,C;KAjBA,C;+FAmBA,yB;MAAA,4F;MA9qEI,8D;MA8qEJ,uC;QAS0B,UAEU,M;QAJhC,YArrEgB,cAAR,iBAAQ,C;QAsrEhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,uBAAI,YAAJ,EAAL,oBAAJ,Q;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,cAAJ,EAAL,sBAAJ,UAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;+FAgBA,yB;MAAA,4F;MAtrEI,8D;MASrEJ,uC;QAS0B,UAEU,M;QAJhC,YA7rEgB,cAAR,iBAAQ,C;QA8rEhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,uBAAI,YAAJ,EAAL,oBAAJ,Q;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,cAAJ,EAAL,sBAAJ,UAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;+FAgBA,yB;MAAA,4F;MA9rEI,8D;MA8rEJ,uC;QAS0B,UAEU,M;QAJhC,YArsEgB,cAAR,iBAAQ,C;QAssEhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,uBAAI,YAAJ,EAAL,oBAAJ,Q;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,cAAJ,EAAL,sBAAJ,UAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;+FAgBA,yB;MAAA,4F;MATsEI,8D;MAssEJ,uC;QAS0B,UAEU,M;QAJhC,YA7sEgB,cAAR,iBAAQ,C;QA8sEhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,uBAAI,YAAJ,EAAL,oBAAJ,Q;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,uBAAI,cAAJ,EAAL,sBAAJ,UAAV,EAawB,WAAxB,C;;QAEIB,OAAO,W;MACX,C;KAdA,C;6GAgBA,yB;MAAA,4F;MA9uEI,8D;MA8uEJ,uC;QAY0B,Q;QAFtB,YAxvEgB,cAAR,iBAAQ,C;QAYvEhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,uBAAI,YAAJ,EAAL,oBAAJ,Q;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,sBAAL,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAIBA,C;6GAoBA,yB;MAAA,4F;MA1vEI,8D;MA0vEJ,uC;QAY0B,Q;QAFtB,YApwEgB,cAAR,iBAAQ,C;QAqwEhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,uBAAI,YAAJ,EAAL,oBAAJ,Q;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,sBAAL,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAIBA,C;6GAoBA,yB;MAAA,4F;MATwEI,8D;MASwEJ,uC;QAY0B,Q;QAFtB,YAhxEgB,cAAR,iBAAQ,C;QAixEhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,uBAAI,YAAJ,EAAL,oBAAJ,Q;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,sBAAL,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAIBA,C;6GAoBA,yB;MAAA,4F;MA1xEI,8D;MAkxEJ,uC;QAY0B,Q;QAFtB,YA5xEgB,cAAR,iBAAQ,C;QA6xEhB,IAAI,QAAQ,CAAZ,C;UAAe,MAAM,mCAA8B,+BAA9B,C;QACrB,kBAakB,uBAAI,YAAJ,EAAL,oBAAJ,Q;QACIB,OAAO,SAAS,CAAhB,C;UACI,cAAc,UAAU,KAAV,EAAiB,sBAAL,KAAJ,CAAjB,EAA6B,WAA7B,C;UACd,qB;;QAEJ,OAAO,W;MACX,C;KAIBA,C;6GAoBA,yB;MAAA,wB;MAAA,sC;QAQoB,Q;QADhB,UAAgB,W;QACA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,MjC1/KiD,SiC0/KjD,GjC1/K2D,KAAK,GiC0/KzD,SAAS,OAAT,CjC1/KoE,KAAK,IAAf,C;;QiC4/KrD,OAAO,G;MACX,C;KAZA,C;mFAcA,yB;MAAA,wB;MAAA,sC;QAQoB,Q;QADhB,UAAgB,W;QACA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,MjCxgLiD,SiCwgLiD,GjCxlL2D,KAAK,GiCwgLzD,SAAS,OAAT,CjCxlLoE,KAAK,IAAf,C;;QiC0gLrD,OAAO,G;MACX,C;KAZA,C;mFAcA,yB;MAAA,wB;MAAA,sC;QAQoB,Q;QADhB,UAAgB,W;QACA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,MjCt hLiD,SiCshLiD,GjCthL2D,KAAK,GiCshLzD,SAAS,OAAT,CjCthLoE,KAAK,IAAf,C;;QiCwhLrD,OAAO,G;MACX,C;KAZA,C;mFAcA,yB;MAAA,wB;MAAA,sC;QAQoB,Q;QADhB,UAAgB,W;QACA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,MjCpiLiD,SiCoiLiD,GjCpiL2D,KAAK,GiCoiLzD,SAAS,OAAT,CjCpiLoE,KAAK,I

AAf,C;;QiCsiLrD,OAAO,G;MACX,C;KAZA,C;;MAsoBoB,Q;MADhB,UAAkB,G;MACF,2B;MAAhB,OAAgB,cA
AhB,C;QAAgB,yB;QACZ,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;IACX,C;;MAUoB,Q;MADhB,UAAkB,G;M
ACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;IACX,C;;MA
UoB,Q;MADhB,UAAkB,G;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,OAAO,SAAS,OAAT,C;;M
AEX,OAAO,G;IACX,C;;MAUoB,Q;MADhB,UAAkB,G;MACF,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAC
Z,OAAO,SAAS,OAAT,C;;MAEX,OAAO,G;IACX,C;;MA6DI,WIBljLO,MAAK,KkBkJLK,cIBljLL,EkBggLD,KA
kDkB,OIBljLjB,C;MkBmjLZ,WAAW,iBAaA,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WA
rDqB,GAqDP,sBAAK,CAAL,CARDO,EAAAnB,KAqDqB,CAAM,CAAN,CARDF,CAqDrB,C;;MARdT,OAuDO,I;I
AtDX,C;;MAoEI,WIBrkLO,MAAK,KkBqkLK,cIBrkLL,EkB4gLD,KAYdKB,OIBrkLjB,C;MkBskLZ,WAAW,iBA
Aa,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WA5DqB,GA4DP,sBAAK,CAAL,CA5DO,EA
AnB,KA4DqB,CAAM,CAAN,CA5DF,CA4DrB,C;;MA5DT,OA8DO,I;IA7DX,C;;MA2EI,WIBxILO,MAAK,KkB
wlLK,cIBxILL,EkBwhLD,KAgEkB,OIBxILjB,C;MkBylLZ,WAAW,iBAaA,IAAb,C;MACX,aAAU,CAAV,MAAk
B,IAAIB,M;QACI,IAAK,WAnEqB,GAmEP,sBAAK,CAAL,CAnEO,EAAAnB,KAmEqB,CAAM,CAAN,CAnEF,C
AmErB,C;;ManET,OAqEO,I;IAPeX,C;;MAkFI,WIB3mLO,MAAK,KkB2mLK,cIB3mLL,EkBoiLD,KAuEkB,OIB
3mLjB,C;MkB4mLZ,WAAW,iBAaA,IAAb,C;MACX,aAAU,CAAV,MAAkB,IAAIB,M;QACI,IAAK,WA1EqB,G
A0EP,sBAAK,CAAL,CA1EO,EAAAnB,KA0EqB,CAAM,CAAN,CA1EF,CA0ErB,C;;MA1ET,OA4EO,I;IA3EX,C;
+EAEA,yB;MAAA,gE;MIB9iLA,iB;MkB8iLA,8C;QAWI,WIBljLO,MAAK,KkBkJLK,cIBljLL,EkBkJLW,KAAM,
OIBljLjB,C;QkBmjLZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAA
U,sBAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAjBA,C;+EAm
BA,yB;MAAA,gE;MIBjkLA,iB;MkBikLA,8C;QAWI,WIBrkLO,MAAK,KkBqkLK,cIBrkLL,EkBqkLW,KAAM,OI
BrkLjB,C;QkBskLZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,
sBAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAjBA,C;+EAmBA
,yB;MAAA,gE;MIBplLA,iB;MkBolLA,8C;QAWI,WIBxILO,MAAK,KkBwlLK,cIBxILL,EkBwlLW,KAAM,OIBxl
LjB,C;QkBylLZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,sBA
AK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAjBA,C;+EAmBA,yB;
MAAA,gE;MIBvmLA,iB;MkBumLA,8C;QAWI,WIB3mLO,MAAK,KkB2mLK,cIB3mLL,EkB2mLW,KAAM,OIB
3mLjB,C;QkB4mLZ,WAAW,eAAa,IAAb,C;QACX,aAAU,CAAV,MAAkB,IAAIB,M;UACI,IAAK,WAAI,UAAU,
sBAAK,CAAL,CAAV,EAAmB,MAAM,CAAN,CAAnB,CAAJ,C;;QAET,OAAO,I;MACX,C;KAjBA,C;;MAiFoB,
gB;MAHhB,gBAAGB,c;MACHB,WAAW,iBIB/qLJ,MAAK,KkB+qLwB,wBAnDzB,KAmDyB,EAAwB,EAAXB,C
IB/qLxB,EkB+qLqD,SIB/qLrD,CkB+qLD,C;MACX,QAAQ,C;MACQ,OArdL,KAqDK,W;MAAhB,OAAgB,cAA
hB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAvDqB,GAuDP,uBAAK,UAAAL,EA
AK,kBAAL,UAvDO,EAuDI,OAvdJ,CAuDrB,C;;MAvDT,OAYDO,I;IAxDX,C;;MAyEoB,gB;MAHhB,gBAAGB,c
;MACHB,WAAW,iBIBpsLJ,MAAK,KkBosLwB,wBA5DzB,KA4DyB,EAAwB,EAAXB,CIBpsLxB,EkBosLqD,SIB
psLrD,CkBosLD,C;MACX,QAAQ,C;MACQ,OA9DL,KA8DK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,
IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAhEqB,GAgEP,uBAAK,UAAAL,EAAK,kBAAL,UAhEO,EAGe
I,OAHEJ,CAGErB,C;;MAhET,OAkEO,I;IAjEX,C;;MAkFoB,gB;MAHhB,gBAAGB,c;MACHB,WAAW,iBIBzLJ,M
AAK,KkBylLwB,wBArEzB,KAqEyB,EAAwB,EAAXB,CIBzLxB,EkBytLqD,SIBzLrD,CkBylLD,C;MACX,QAA
Q,C;MACQ,OAveL,KAuEK,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K
;QACpB,IAAK,WAZEqB,GAYEP,uBAAK,UAAAL,EAAK,kBAAL,UAZEO,EAYEI,OAzEJ,CAYErB,C;;MAzET,OA
2EO,I;IA1EX,C;;MA2FoB,gB;MAHhB,gBAAGB,c;MACHB,WAAW,iBIB9uLJ,MAAK,KkB8uLwB,wBA9EzB,K
A8EyB,EAAwB,EAAXB,CIB9uLxB,EkB8uLqD,SIB9uLrD,CkB8uLD,C;MACX,QAAQ,C;MACQ,OAFL,KAqF
K,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,KAAK,SAAT,C;UAAoB,K;QACpB,IAAK,WAlFqB,G
AkFP,uBAAK,UAAAL,EAAK,kBAAL,UAlFO,EAkFI,OAIFJ,CakFrB,C;;MAIFT,OAoFO,I;IANFX,C;+EAEA,yB;
MAAA,kF;MAAA,gE;MIB1qLA,iB;MkB0qLA,8C;QAcOB,UAEY,M;QAL5B,gBAAGB,c;QACHB,WAAW,eIB/qL
J,MAAK,KkB+qLwB,wBAAN,KAAM,EAAwB,EAAXB,CIB/qLxB,EkB+qLqD,SIB/qLrD,CkB+qLD,C;QACX,Q
AAQ,C;QACQ,uB;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAAI,KAAK,SAAT,C;YAAoB,K;UACpB,IAA
K,WAAI,UAAU,uBAAK,UAAAL,EAAK,kBAAL,UAAV,EAAqB,OAARb,CAAJ,C;;QAET,OAAO,I;MACX,C;KA
nBA,C;+EAqBA,yB;MAAA,kF;MAAA,gE;MIB/rLA,iB;MkB+rLA,8C;QAcOB,UAEY,M;QAL5B,gBAAGB,c;QA

ChB, WAAW, elBpsLJ, MAAK, KkBosLwB, wBAAN, KAAM, EAAwB, EAAxB, ClBpsLxB, EkBosLqD, SIBpsLrD, Ck
BosLD, C; QACX, QAAQ, C; QACQ, uB; QAAhB, OAAgB, cAAhB, C; UAAgB, yB; UACZ, IAAI, KAAK, SAAT, C; YAA
oB, K; UACpB, IAAK, WAAI, UAAU, uBAAK, UAAL, EAAK, kBAAL, UAAV, EAAqB, OAArB, CAAJ, C;; QAET, OA
AO, I; MACX, C; KAnBA, C; +EAqBA, yB; MAAA, kF; MAAA, gE; MIBptLA, iB; MkBotLA, 8C; QacoB, UAEY, M; QAL
5B, gBAAgB, c; QACHB, WAAW, elBztLJ, MAAK, KkBytLwB, wBAAN, KAAM, EAAwB, EAAxB, ClBztLxB, EkBytL
qD, SIBztLrD, CkBytLD, C; QACX, QAAQ, C; QACQ, uB; QAAhB, OAAgB, cAAhB, C; UAAgB, yB; UACZ, IAAI, KAA
K, SAAT, C; YAAoB, K; UACpB, IAAK, WAAI, UAAU, uBAAK, UAAL, EAAK, kBAAL, UAAV, EAAqB, OAArB, CA
AJ, C;; QAET, OAAO, I; MACX, C; KAnBA, C; 8EAqBA, yB; MAAA, kF; MAAA, gE; MIBzuLA, iB; MkByuLA, 8C; Qaco
B, UAEY, M; QAL5B, gBAAgB, c; QACHB, WAAW, elB9uLJ, MAAK, KkB8uLwB, wBAAN, KAAM, EAAwB, EAAxB
, ClB9uLxB, EkB8uLqD, SIB9uLrD, CkB8uLD, C; QACX, QAAQ, C; QACQ, uB; QAAhB, OAAgB, cAAhB, C; UAAgB, y
B; UACZ, IAAI, KAAK, SAAT, C; YAAoB, K; UACpB, IAAK, WAAI, UAAU, uBAAK, UAAL, EAAK, kBAAL, UAAV,
EAAqB, OAArB, CAAJ, C;; QAET, OAAO, I; MACX, C; KAnBA, C;; MAgFI, WIBlzLO, MAAK, KkBkzLK, clBlzLL, Ek
BgwLD, KAKdKB, KIBlzLjB, C; MkBmzLZ, WAAW, iBAaA, IAAb, C; MACX, aAAU, CAAV, MAaKB, IAAIB, M; QAC
I, IAAK, WArDqB, GAqDP, sBAAK, CAAL, CArDO, EAAnB, KAqDqB, aAAM, CAAN, CArDF, CAqDrB, C;; MARDT,
OAuDO, I; IAtdX, C;; MAoEI, WIBr0LO, MAAK, KkBq0LK, clBr0LL, EkB4wLD, KAyDkB, KIBr0LjB, C; MkBs0LZ,
WAAW, iBAaA, IAAb, C; MACX, aAAU, CAAV, MAaKB, IAAIB, M; QACI, IAAK, WA5DqB, GA4DP, sBAAK, CAAL
, CA5DO, EAAnB, KA4DqB, aAAM, CAAN, CA5DF, CA4DrB, C;; MA5DT, OA8DO, I; IA7DX, C;; MA2EI, WIBx1LO,
MAAK, KkBw1LK, clBx1LL, EkBwxLD, KAqEkB, KIBx1LjB, C; MkBy1LZ, WAAW, iBAaA, IAAb, C; MACX, aAAU,
CAAV, MAaKB, IAAIB, M; QACI, IAAK, WAnEqB, GAmEP, sBAAK, CAAL, CAnEO, EAAnB, KAmEqB, aAAM, CA
AN, CAnEF, CAmErB, C;; MAnET, OAqEO, I; IApEX, C;; MAkFI, WIB32LO, MAAK, KkB22LK, clB32LL, EkBoyLD, K
AuEkB, KIB32LjB, C; MkB42LZ, WAAW, iBAaA, IAAb, C; MACX, aAAU, CAAV, MAaKB, IAAIB, M; QACI, IAAK,
WA1EqB, GA0EP, sBAAK, CAAL, CA1EO, EAAnB, KA0EqB, aAAM, CAAN, CA1EF, CA0ErB, C;; MA1ET, OA4EO, I
; IA3EX, C; +EAEA, yB; MAAA, gE; MIB9yLA, iB; MkB8yLA, 8C; QAWI, WIBlzLO, MAAK, KkBkzLK, clBlzLL, EkBk
zLW, KAAM, KIBlzLjB, C; QkBmzLZ, WAAW, eAAa, IAAb, C; QACX, aAAU, CAAV, MAaKB, IAAIB, M; UACI, IAA
K, WAAI, UAAU, sBAAK, CAAL, CAAV, EAAM, kBAAM, CAAN, CAAAnB, CAAJ, C;; QAET, OAAO, I; MACX, C; K
AjBA, C; +EAmBA, yB; MAAA, gE; MIBj0LA, iB; MkBi0LA, 8C; QAWI, WIBr0LO, MAAK, KkBq0LK, clBr0LL, EkBq
0LW, KAAM, KIBr0LjB, C; QkB0LZ, WAAW, eAAa, IAAb, C; QACX, aAAU, CAAV, MAaKB, IAAIB, M; UACI, IAA
K, WAAI, UAAU, sBAAK, CAAL, CAAV, EAAM, kBAAM, CAAN, CAAAnB, CAAJ, C;; QAET, OAAO, I; MACX, C; K
AjBA, C; +EAmBA, yB; MAAA, gE; MIBp1LA, iB; MkBo1LA, 8C; QAWI, WIBx1LO, MAAK, KkBw1LK, clBx1LL, Ek
Bw1LW, KAAM, KIBx1LjB, C; QkB1LZ, WAAW, eAAa, IAAb, C; QACX, aAAU, CAAV, MAaKB, IAAIB, M; UACI, I
AAK, WAAI, UAAU, sBAAK, CAAL, CAAV, EAAM, kBAAM, CAAN, CAAAnB, CAAJ, C;; QAET, OAAO, I; MACX, C
; KAJBA, C; +EAmBA, yB; MAAA, gE; MIBv2LA, iB; MkBu2LA, 8C; QAWI, WIB32LO, MAAK, KkB22LK, clB32LL, E
kB22LW, KAAM, KIB32LjB, C; QkB42LZ, WAAW, eAAa, IAAb, C; QACX, aAAU, CAAV, MAaKB, IAAIB, M; UACI, I
AAK, WAAI, UAAU, sBAAK, CAAL, CAAV, EAAM, kBAAM, CAAN, CAAAnB, CAAJ, C;; QAET, OAAO, I; MACX, C
; KAJBA, C;; MA2BoB, Q; MADhB, UAAgB, W; MACHB, wBAAgB, SAAhB, gB; QAAgB, cAAA, SAAhB, M; QACI, Mj
Ct+LiD, SiCs+LjD, GjCt+L2D, KAAK, GiCs+LzD, OjCt+LoE, KAAX, IAaf, C;; MiCw+LrD, OAAO, G; IACX, C;; MAU
oB, Q; MADhB, UAAiB, 2B; MACjB, wBAAgB, SAAhB, gB; QAAgB, cAAA, SAAhB, M; QACI, MFj/LmD, UEi/LnD, G
Fj/L8D, KAAK, KEi/L5D, OFj/LuE, KAAX, CAAhB, C;; MEm/LvD, OAAO, G; IACX, C;; MAUoB, Q; MADhB, UAAgB,
W; MACHB, wBAAgB, SAAhB, gB; QAAgB, cAAA, SAAhB, M; QACI, MjClgMiD, SiCkgMjD, GjClgM2D, KAAK, GA
AW, CDgJ5C, SkCk3LxB, OICl3LkC, KAAL, GAAiB, GAAtB, CChJ4C, MAAX, IAaf, C;; MiCogMrD, OAAO, G; IACX
, C;; MAUoB, Q; MADhB, UAAgB, W; MACHB, wBAAgB, SAAhB, gB; QAAgB, cAAA, SAAhB, M; QACI, MjChhMiD,
SiCghMjD, GjChhM2D, KAAK, GAAW, CCiJ5C, SgC+3LxB, OhC/3LkC, KAAL, GAAiB, KAAtB, CDjJ4C, MAAX, IA
Af, C;; MiCkhMrD, OAAO, G; IACX, C; +EAEA, yB; MAAA, 0C; MjCp0LA, 6B; MiCo0LA, 4B; QAOI, OjCj0LmC, ciCi0
LpB, IAAR, iBAAQ, CjCj0LoB, C; MiCk0LvC, C; KARA, C; +EAUA, yB; MAAA, 0C; MF/zLA, +B; ME+zLA, 4B; QAOI,
OF5zLsC, eE4zLvB, IAAR, iBAAQ, CF5zLuB, C; ME6zL1C, C; KARA, C; +EAUA, yB; MAAA, sC; MA5hBA, wB; MA4
hBA, iBAOiB, yB; QICz6Lb, 6B;; UkCy6Le, OICh6LoB, ckCg6LpB, EICb6L8B, KAAL, GAAiB, GAAtB, C; QkCg6LR, C
; OAAAd, C; MAPjB, 4B; QApHBoB, Q; QADhB, UAAgB, W; QACA, 2B; QAAhB, OAAgB, cAAhB, C; UAAgB, yB; UACZ
, MjCthLiD, SiCshLjD, GjCthL2D, KAAK, GAAW, CDgJ5C, SkCs4Kf, OICt4KyB, KAAL, GAAiB, GAAtB, CChJ4C, M

AAX,IAAf,C;;QiCgjMrD,OAxhBO,G;MAyhBX,C;KARA,C;+EAUA,yB;MAAA,sC;MAxhBA,wB;MAwhBA,iBA
OiB,yB;QhCl7Lb,6B;;UgCk7Le,OhCz6LoB,cgCy6LpB,EhCz6L8B,KAAL,GAAiB,KAAtB,C;QgCy6LR,C;OAA
C;MAPjB,4B;QAhhBoB,Q;QADhB,UAAgB,W;QACA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UACZ,MjCpi
LiD,SiCoiLjD,GjCpiL2D,KAAK,GAAW,CCiJ5C,SgCm5Kf,OhCn5KyB,KAAL,GAAiB,KAAtB,CDjJ4C,MAAX,I
AAf,C;;QiC0jMrD,OAphBO,G;MAqhBX,C;KARA,C;;MCvmMoB,UACL,M;MAHX,aAAa,gBAAW,cAAX,C;M
ACb,YAAY,C;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,oBAAO,cAAP,EAAO,sBAAP,WAAkB,
OAAIB,C;;MACJ,OAAO,M;IACX,C;;MAUoB,UACL,M;MAHX,aAAa,eAAU,cAAV,C;MACb,YAAY,C;MACI,2
B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,oBAAO,cAAP,EAAO,sBAAP,WAAkB,OAAIB,C;;MACJ,OAA
O,M;IACX,C;;MAUoB,UACL,M;MAHX,aAAa,gBAAW,cAAX,C;MACb,YAAY,C;MACI,2B;MAAhB,OAAgB,c
AAhB,C;QAAgB,yB;QACZ,oBAAO,cAAP,EAAO,sBAAP,WAAkB,OAAIB,C;;MACJ,OAAO,M;IACX,C;;MAUo
B,UACL,M;MAHX,aAAa,iBAAY,cAAZ,C;MACb,YAAY,C;MACI,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;Q
ACZ,oBAAO,cAAP,EAAO,sBAAP,WAAkB,OAAIB,C;;MACJ,OAAO,M;IACX,C;;MAUoB,Q;MADhB,UAAgB,
W;MACA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MICDiD,SkCCjD,GICD2D,KAAK,GkCCzD,OICD
oE,KAAx,IAAf,C;;MkCGrD,OAAO,G;IACX,C;;MAUoB,Q;MADhB,UAAiB,2B;MACD,2B;MAAhB,OAAgB,cA
AhB,C;QAAgB,yB;QACZ,MHZmD,UGYnD,GHZ8D,KAAK,KGy5D,OHZuE,KAAx,CAAhB,C;;MGcvD,OAAO
,G;IACX,C;;MAUoB,Q;MADhB,UAAgB,W;MACA,2B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QACZ,MIC7BiD
,SkC6BjD,GIC7B2D,KAAK,GAAW,CDgJ5C,SmCnHxB,OnCmHkC,KAAL,GAAiB,GAAtB,CChJ4C,MAAX,IA
Af,C;;MkC+BrD,OAAO,G;IACX,C;;MAUoB,Q;MADhB,UAAgB,W;MACA,2B;MAAhB,OAAgB,cAAhB,C;QA
AgB,yB;QACZ,MIC3CiD,SkC2CjD,GIC3C2D,KAAK,GAAW,CCiJ5C,SiCtGxB,OjCsGkC,KAAL,GAAiB,KAAtB
,CDjJ4C,MAAX,IAAf,C;;MkC6CrD,OAAO,G;IACX,C;;MctGI,OnCsCkE,YmCtCvD,CnCsCwE,KAAjB,EmCtCl
D,CnCsC+E,KAA7B,CmCtCvD,KAAJ,GAAY,CAAZ,GAAMB,C;IAC9B,C;;MAQI,OJqCmE,aIrcxD,CJqC0E,KA
AlB,ElrCnD,CJqCiF,KAA9B,CiRcxD,KAAJ,GAAY,CAAZ,GAAMB,C;IAC9B,C;;MAQI,OpCIgF,0BoCjRE,CpCo
J2B,KAAL,GAAiB,GAhJ8B,EoCjHE,CpCoJsB,KAAL,GAAiB,GAhJ8B,CoCjRE,KAAJ,GAAY,CAAZ,GAAMB,C;
IAC9B,C;;MAQI,OICgIF,0BkCHtE,CIC4I2B,KAAL,GAAiB,KazI+B,EkCHjE,CIC4IsB,KAAL,GAAiB,KazI+B,
CkCHtE,KAAJ,GAAY,CAAZ,GAAMB,C;IAC9B,C;kFAEA,yB;MAAA,8C;MAAA,0B;QAOI,OAAO,MAAM,CA
AN,EAAS,MAAM,CAAN,EAAS,CAAT,CAAT,C;MACX,C;KARA,C;kFAUA,yB;MAAA,8C;MAAA,0B;QAOI,
OAAO,MAAM,CAAN,EAAS,MAAM,CAAN,EAAS,CAAT,CAAT,C;MACX,C;KARA,C;kFAUA,yB;MAAA,8C;
MAAA,0B;QAOI,OAAO,MAAM,CAAN,EAAS,MAAM,CAAN,EAAS,CAAT,CAAT,C;MACX,C;KARA,C;kFA
UA,yB;MAAA,8C;MAAA,0B;QAOI,OAAO,MAAM,CAAN,EAAS,MAAM,CAAN,EAAS,CAAT,CAAT,C;MAC
X,C;KARA,C;;MAGBI,OnCtCkE,YmCsCvD,CnCtCwE,KAAjB,EmCsCID,CnCtC+E,KAA7B,CmCsCvD,KAAJ,G
AAY,CAAZ,GAAMB,C;IAC9B,C;;MAQI,OJvCmE,aIuCXD,CJvC0E,KAAIB,EluCnD,CJvCiF,KAA9B,CluCXD,K
AAJ,GAAY,CAAZ,GAAMB,C;IAC9B,C;;MAQI,OpCxEgF,0BoCwErE,CpCwE2B,KAAL,GAAiB,GAhJ8B,EoCw
EhE,CpCwEsB,KAAL,GAAiB,GAhJ8B,CoCwErE,KAAJ,GAAY,CAAZ,GAAMB,C;IAC9B,C;;MAQI,OICzEiF,0
BkCyEtE,CICgE2B,KAAL,GAAiB,KazI+B,EkCyEjE,CICgEsB,KAAL,GAAiB,KazI+B,CkCyEtE,KAAJ,GAAY,
CAAZ,GAAMB,C;IAC9B,C;kFAEA,yB;MAAA,8C;MAAA,0B;QAOI,OAAO,MAAM,CAAN,EAAS,MAAM,CA
AN,EAAS,CAAT,CAAT,C;MACX,C;KARA,C;kFAUA,yB;MAAA,8C;MAAA,0B;QAOI,OAAO,MAAM,CAAN,
EAAS,MAAM,CAAN,EAAS,CAAT,CAAT,C;MACX,C;KARA,C;kFAUA,yB;MAAA,8C;MAAA,0B;QAOI,OAA
O,MAAM,CAAN,EAAS,MAAM,CAAN,EAAS,CAAT,CAAT,C;MACX,C;KARA,C;kFAUA,yB;MAAA,8C;MA
AA,0B;QAOI,OAAO,MAAM,CAAN,EAAS,MAAM,CAAN,EAAS,CAAT,CAAT,C;MACX,C;KARA,C;gFC9IA,
yB;MAAA,mC;MAAA,2C;MAAA,4B;QASI,OAAO,kBAAO,cAAP,C;MACX,C;KAVA,C;;QAqBQ,OAAc,WAAP,MAAO,EA
AS,SAAT,C;;QACHB,+C;UACE,MAAM,2BAAuB,CAAE,QAAzB,C;;UAHV,O;;IAKJ,C;;QAWQ,OAAc,YAAP
,MAAO,EAAU,SAAV,C;;QACHB,+C;UACE,MAAM,2BAAuB,CAAE,QAAzB,C;;UAHV,O;;IAKJ,C;;MAWI,O
AAO,WAAW,IAAX,IAAmB,2BAAS,OAAT,C;IAC9B,C;;MAWI,OAAO,WAAW,IAAX,IAAmB,2BAAS,OAAT,
C;IAC9B,C;;MAQI,OAAO,2BrC0H4B,SqC1HnB,KrC0H6B,KAAL,GAAiB,GAAtB,CqC1H5B,C;IACX,C;;MAQI
,OAAO,2BrC2H8B,UAAW,oBqC3HhC,KrC2H2B,KAAK,CAAL,UAAO,CqC3H9B,C;IACX,C;;MAQI,OAAO,2B
pC0H8B,UAAW,oBoC1HhC,KpC0H2B,KAAK,CAAL,iBAAN,CoC1H9B,C;IACX,C;;MAQY,Q;MAAD,cAAC,O
LwB4C,UKxB5C,KLwBkD,yBKxBxC,ELwBwC,CAAN,CKxB7C,wBAA8B,2BAA9B,Q;MAAA,W;QAAqC,oCp

CsLR,SoCtLiB,KL+GIB,K/BuEW,QA AV,CoCtLQ,C;;MAA5C,a;IACJ,C;;MAQI,OAAO,2BnCuF4B,SmCvFnB,K nCuF6B,KAAL,GAAiB,KAAtB,CmCvF5B,C;IACX,C;;MAQI,OAAO,2BnCuF8B,UAAW,oBmCxFhC,KnCwF2B ,KAAK,CAAL,YAAN,CmCxF9B,C;IACX,C;;MAWI,OAAO,uCAA gB,yBrCiEY,SqCjEl,SrCiEM,KAAL,GAAiB, GAAtB,CqCjEZ,ErCiEY,SqCjEmB,ErCiET,KAAL,GAAiB,GAAtB,CqCjEZ,EAA4C,EAA5C,C;IAC3B,C;;MAWI ,OAAO,uCAA gB,yBAA gB,SAAhB,EAA sB,EAA tB,EAA0B,EAA1B,C;IAC3B,C;;MAWI,OAAO,wCAAiB,yBAA gB,SAAhB,EAA sB,EAA tB,M;IAC5B,C;;MAWI,OAAO,uCAA gB,yBnC8BY,SmC9BI,SnC8BM,KAAL,GAAiB,K AA tB,CmC9BZ,EnC8BY,SmC9BmB,EnC8BT,KAAL,GAAiB,KAA tB,CmC9BZ,EAA4C,EAA5C,C;IAC3B,C;;M AQI,OAAO,uCAA gB,yBAA gB,cAAhB,EAA sB,eAA tB,EAA6B,CAAC,cAAD,IAA7B,C;IAC3B,C;;MAQI,OAAO ,wCAAiB,yBAA gB,cAAhB,EAA sB,eAA tB,EAA8B,cAAD,aaa7B,C;IAC5B,C;;MAQI,oBAAoB,OAAO,CAA3B, EAA8B,IAA9B,C;MACA,OAAO,uCAA gB,yBAA gB,eAAhB,EAAuB,cAAvB,EAAiC,SAAK,KAAL,GAA Y,CAA hB,GAAmB,IAAnB,GAA6B,CAAC,IAAD,IAA1D,C;IAC3B,C;;MAQI,oBAAoB,kBAAO,CAA3B,EAA8B,IAA9 B,C;MACA,OAAO,wCAAiB,yBAA gB,eAAhB,EAAuB,cAAvB,EAAiC,SAAK,KAAL,cAA Y,CAAhB,GAAmB,I AAnB,GAA8B,IAAD,aaa1D,C;IAC5B,C;;MAUI,IrCtMgF,0BqCsM5E,ErCtDkC,KAAL,GAAiB,GAhJ8B,EqCsM tE,6BAAM,UrCtDsB,KAAL,GAAiB,GAhJ8B,CqCsM5E,KAAJ,C;QAA2B,OAAO,iCAAU,M;MACHC,WrCrBuB, SqCqB5B,SrCrBsC,KAAL,GAAiB,GAA tB,C;MqCqBV,YAAK,W;MAA9B,OpCnG6D,oBArDP,SAAU,CDmI7B, SqCqBV,ErCrBoB,KAAL,GAAiB,GAA tB,CCnI6B,MAAK,GDAK,KCAO,KAAZ,IAAf,CAqDO,C;IoCoGjE,C;;M AUUI,IpCImkE,YoCkM9D,EpCim+E,KAAjB,EoCkMxD,4BAAK,UpCimGf,KAA7B,CoCkM9D,KAAJ,C;QAA0B, OAAO,iCAAU,M;MAC3C,OpC/G6D,coC+GtD,SpC/GsD,EA rDP,SoCoKtC,EpCpKgD,KAAK,GAA Y,CoCoK5D, WpCpK4D,MAAZ,IAAf,CAqDO,C;IoCgHjE,C;;MAUI,ILtMmE,aKsM/D,ELtMiF,KAAIB,EKsMzD,6BAAM,ULt MiF,KAA9B,CKsM/D,KAAJ,C;QAA2B,OAAO,kCAAW,M;MAC7C,OL3H+D,iBK2HxD,SL3HwD,EAIDP,UK6 KxC,EL7KmD,KAAK,UAA Y,C/BkJ/C,UAAW,oBAAAL,CoC2BtB,WpC3BsB,MAAK,CAAL,iBAAN,C+BIJ+C,M AAZ,CAAhB,CakDO,C;IK4HnE,C;;MAUI,InClOIF,0BmCkO7E,EnCzFkC,KAAL,GAAiB,KAZI+B,EmCkOvE,8 BAAO,UnCzFqB,KAAL,GAAiB,KAZI+B,CmCkO7E,KAAJ,C;QAA4B,OAAO,iCAAU,M;MACjC,WnCxDuB,Sm CwD5B,SnCxDsC,KAAL,GAAiB,KAA tB,C;MmCwDV,YAAK,W;MAA9B,OpCvI6D,oBArDP,SAAU,CCoI7B,S mCwDV,EnCxDob,KAAL,GAAiB,KAA tB,CDpI6B,MAAK,GCAK,KDAO,KAAZ,IAAf,CAqDO,C;IoCwIjE,C;; MAYI,OpCxoKE,YoCwOvD,SpCxoWE,KAAjB,EoCwOhD,YpCxO6E,KAA7B,CoCwOvD,IAAJ,GAAyB,YAAz B,GAA2C,S;IACtD,C;;MAYI,OL7OmE,aK6OxD,SL7O0E,KAAIB,EK6OjD,YL7O+E,KAA9B,CK6OxD,IAAJ,G AAyB,YAAzB,GAA2C,S;IACtD,C;;MAYI,OrClRgF,0BqCkRrE,SrClI2B,KAAL,GAAiB,GAhJ8B,EqCkR9D,YrCl IoB,KAAL,GAAiB,GAhJ8B,CqCkRrE,IAAJ,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAYI,OnCvRiF,0BmCuRtE,S nC9I2B,KAAL,GAAiB,KAZI+B,EmCuR/D,YnC9IoB,KAAL,GAAiB,KAZI+B,CmCuRtE,IAAJ,GAAyB,YAAzB, GAA2C,S;IACtD,C;;MAYI,OpC5RkE,YoC4RvD,SpC5RwE,KAAjB,EoC4RhD,YpC5R6E,KAA7B,CoC4RvD,IA AJ,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAYI,OLjSmE,aKiSxD,SLjS0E,KAAIB,EKiSjD,YLjS+E,KAA9B,CKi SxD,IAAJ,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAYI,OrCtUgF,0BqCsUrE,SrCtL2B,KAAL,GAAiB,GAhJ8B,E qCsU9D,YrCtLoB,KAAL,GAAiB,GAhJ8B,CqCsUrE,IAAJ,GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAYI,OnC3Ui F,0BmC2UtE,SnCIM2B,KAAL,GAAiB,KAZI+B,EmC2U/D,YnCIMO b,KAAL,GAAiB,KAZI+B,CmC2UtE,IAAJ, GAAyB,YAAzB,GAA2C,S;IACtD,C;;MAYI,IpChVke,YoCgV9D,YpChV+E,KAAjB,EoCgV/C,YpChV4E,KAA7 B,CoCgV9D,IAAJ,C;QAAiC,MAAM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IpCjVke, YoCiV9D,SpCjV+E,KAAjB,EoCiVvD,YpCjVoF,KAA7B,CoCiV9D,IAAJ,C;QAAyB,OAAO,Y;MACHC,IpCIVkE ,YoCkV9D,SpCIV+E,KAAjB,EoCkVvD,YpCIVoF,KAA7B,CoCkV9D,IAAJ,C;QAAyB,OAAO,Y;MACHC,OAAO ,S;IACX,C;;MAYI,ILxVmE,aKwV/D,YLxViF,KAAIB,EKwVhD,YLxV8E,KAA9B,CKwV/D,IAAJ,C;QAAiC,MA AM,gCAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,ILxVmE,aKyV/D,SLzViF,KAAIB,EKyVx D,YLzVsF,KAA9B,CKyV/D,IAAJ,C;QAAyB,OAAO,Y;MACHC,IL1VmE,aK0V/D,SL1ViF,KAAIB,EK0VxD,YL 1VsF,KAA9B,CK0V/D,IAAJ,C;QAAyB,OAAO,Y;MACHC,OAAO,S;IACX,C;;MAYI,IrChYgF,0BqCgY5E,YrCh PkC,KAAL,GAAiB,GAhJ8B,EqCgY7D,YrChPmB,KAAL,GAAiB,GAhJ8B,CqCgY5E,IAAJ,C;QAAiC,MAAM,g CAAyB,oDAAiD,YAAjD,8BAAoF,YAApF,MAAzB,C;MACvC,IrCjYgF,0BqCiY5E,SrCjPkC,KAAL,GAAiB,GA hJ8B,EqCiYrE,YrCjP2B,KAAL,GAAiB,GAhJ8B,CqCiY5E,IAAJ,C;QAAyB,OAAO,Y;MACHC,IrCIYgF,0BqCkY 5E,SrCIPkC,KAAL,GAAiB,GAhJ8B,EqCkYrE,YrCIP2B,KAAL,GAAiB,GAhJ8B,CqCkY5E,IAAJ,C;QAAyB,OA AO,Y;MACHC,OAAO,S;IACX,C;;MAYI,InCxYiF,0BmCwY7E,YnC/PkC,KAAL,GAAiB,KAZI+B,EmCwY9D,Y

nC/PmB,KAAL,GAAiB,KAZI+B,CmCwY7E,IAAJ,C;QAAiC,MAAM,gCAAYB,oDAAiD,YAAjD,8BAAoF,YAA
pF,MAAZB,C;MACvC,InCzYiF,0BmCyY7E,SnChQkC,KAAL,GAAiB,KAZI+B,EmCyYtE,YnChQ2B,KAAL,GA
AiB,KAZI+B,CmCyY7E,IAAJ,C;QAAyB,OAAO,Y;MACHc,InC1YiF,0BmC0Y7E,SnCjQkC,KAAL,GAAiB,KAZ
I+B,EmC0YtE,YnCjQ2B,KAAL,GAAiB,KAZI+B,CmC0Y7E,IAAJ,C;QAAyB,OAAO,Y;MACHc,OAAO,S;IACX,
C;;MAGBW,Q;MAJP,IAAI,8CAAJ,C;QACI,OAAy,WAAL,SAAK,EAAe,KAAf,C;;MAEhB,IAAI,KAAM,UAAV,
C;QAAqB,MAAM,gCAAYB,4CAAYC,KAAzC,MAAZB,C;MAEvB,IpCrZ8D,YoCqZ9D,SpCrZ+E,KAAjB,EoCqZ
vD,KAAM,MpCrZ8E,KAA7B,CoCqZ9D,K;QAA4B,OAAN,KAAM,M;;QAC5B,IpCtZ8D,YoCsZ9D,SpCtZ+E,K
AAjB,EoCsZvD,KAAM,apCtZ8E,KAA7B,CoCsZ9D,K;UAAmC,OAAN,KAAM,a;;UAC3B,gB;;MAHZ,W;IAKJ,
C;;MAGBW,Q;MAJP,IAAI,8CAAJ,C;QACI,OAAy,WAAL,SAAK,EAAgB,KAAhB,C;;MAEhB,IAAI,KAAM,UA
AV,C;QAAqB,MAAM,gCAAYB,4CAAYC,KAAzC,MAAZB,C;MAEvB,ILla+D,aKka/D,SLlaiF,KAAIB,EKkaxD,
KAAM,MLlagF,KAA9B,CKka/D,K;QAA4B,OAAN,KAAM,M;;QAC5B,ILna+D,aKma/D,SLnaiF,KAAIB,EKmax
D,KAAM,aLlagF,KAA9B,CKma/D,K;UAAmC,OAAN,KAAM,a;;UAC3B,gB;;MAHZ,W;IAKJ,C;;MChdoB,Q;M
ADhB,UAAgB,W;MACA,2B;MAAhB,OAAGB,cAAhB,C;QAAgB,yB;QACZ,MrCmDiD,SqCnDjD,GrCmD2D,K
AAK,GqCnDzD,OrCmDoE,KAAx,IAAf,C;;MqCjDrD,OAAO,G;IACX,C;;MAYoB,Q;MADhB,UAAiB,2B;MAC
D,2B;MAAhB,OAAGB,cAAhB,C;QAAgB,yB;QACZ,MNsCmD,UMtCnD,GNsC8D,KAAK,KMtC5D,ONsCuE,K
AAX,CAAhB,C;;MMPcVd,OAAO,G;IACX,C;;MAYoB,Q;MADhB,UAAgB,W;MACA,2B;MAAhB,OAAGB,cA
AhB,C;QAAgB,yB;QACZ,MrCmBiD,SqCnBjD,GrCmB2D,KAAK,GAAW,CDgJ5C,SsCnKxB,OtCmKkC,KAAL,
GAAiB,GAAtB,CChJ4C,MAAX,IAAf,C;;MqCjBrD,OAAO,G;IACX,C;;MAYoB,Q;MADhB,UAAgB,W;MACA,2
B;MAAhB,OAAGB,cAAhB,C;QAAgB,yB;QACZ,MrCGiD,SqCHjD,GrCG2D,KAAK,GAAW,CCiJ5C,SoCpJxB,O
pCoJkC,KAAL,GAAiB,KAAtB,CDjJ4C,MAAX,IAAf,C;;MqCDrD,OAAO,G;IACX,C;;MC3DA,e;MAAA,iB;
MAAA,uB;IAAA,C;;MAAA,0C;MAAA,C;MAII,kE;MAEA,wF;MAEA,oF;MAEA,wE;MAEA,kE;MAEA,oF;MA
EA,sF;MAEA,8E;MAEA,wE;MAEA,sF;MAEA,uF;MAEA,iE;MAEA,6E;MAEA,iE;MAEA,2E;IAhCJ,C;;MAII,6
B;MAAA,sC;IAAA,C;;MAEA,6B;MAAA,iD;IAAA,C;;MAEA,6B;MAAA,+C;IAAA,C;;MAEA,6B;MAAA,yC;I
AAA,C;;MAEA,6B;MAAA,sC;IAAA,C;;MAEA,6B;MAAA,+C;IAAA,C;;MAEA,6B;MAAA,gD;IAAA,C;;MA
EA,6B;MAAA,4C;IAAA,C;;MAEA,6B;MAAA,yC;IAAA,C;;MAEA,6B;MAAA,gD;IAAA,C;;MAEA,6B;MAA
A,gD;IAAA,C;;MAEA,6B;MAAA,qC;IAAA,C;;MAEA,6B;MAAA,2C;IAAA,C;;MAEA,6B;MAAA,qC;IAAA,
C;;MAEA,6B;MAAA,0C;IAAA,C;;MAhCJ,+oB;IAAA,C;;MAAA,a;aAAA,O;UAAA,2C;aAAA,kB;UAAA,sD;a
AAA,gB;UAAA,oD;aAAA,U;UAAA,8C;aAAA,O;UAAA,2C;aAAA,gB;UAAA,oD;aAAA,iB;UAAA,qD;aAAA,a;
UAAA,iD;aAAA,U;UAAA,8C;aAAA,iB;UAAA,qD;aAAA,iB;UAAA,qD;aAAA,M;UAAA,0C;aAAA,Y;UAAA,g
D;aAAA,M;UAAA,0C;aAAA,W;UAAA,+C;gBAAA,uE;;IAAA,C;;MAqCA,e;MAAA,iB;MAAA,uB;IAAA,C;;M
AAA,6C;MAAA,C;MAMI,0E;MAEA,0E;MAEA,4E;IAVJ,C;;MAMI,gC;MAAA,0C;IAAA,C;;MAEA,gC;MAA
A,0C;IAAA,C;;MAEA,gC;MAAA,2C;IAAA,C;;MAVJ,sI;IAAA,C;;MAAA,a;aAAA,Q;UAAA,+C;aAAA,Q;UA
AA,+C;aAAA,S;UAAA,gD;gBAAA,0E;;IAAA,C;;MAwB+B,oC;IAAD,C;;MAQI,qB;QAAA,iD;MAAA,kB;IAA
D,C;;IAEjC,C;;IAMA,C;;MCOPe,Q;MAAP,OAAO,uE;IACX,C;;MAiEI,OAAO,I;IACX,
C;;ICtYJ,C;;IAQA,C;;MCjBI,UAAU,IAAI,C;MACd,OAAW,OAAO,CAAX,GAAC,GAAd,GAAuB,MAAM,
CAAN,I;IACIC,C;;MAGI,UAAU,SAAI,CAAJ,C;MACV,OAAW,kBAAO,CAAX,GAAC,GAAd,GAAuB,QAAM,C
AAN,C;IACIC,C;;MAII,OAAO,IAAI,IAAI,CAAJ,EAAO,CAAP,IAAY,IAAI,CAAJ,EAAO,CAAP,CAAZ,IAAJ,E
AA2B,CAA3B,C;IACX,C;;MAGI,OAAO,MAAI,MAAI,CAAJ,EAAO,CAAP,WAAY,MAAI,CAAJ,EAAO,CAAP
,CAAZ,CAAJ,EAA2B,CAA3B,C;IACX,C;;MAoBI,WAAO,CAAP,C;QAD2E,OAC3D,SAAS,GAAb,GAakB,GA
AIB,GAA2B,MAAM,iBAAiB,GAAjB,EAA8B,KAAtB,EAA6B,IAA7B,CAAN,I;WACvC,WAAO,CAAP,C;QAF2
E,OAE3D,SAAS,GAAb,GAakB,GAAIB,GAA2B,MAAM,iBAAiB,KAAjB,EAAwB,GAAxB,EAA6B,CAAC,IAA
D,IAA7B,CAAN,I;;QAC/B,MAAA,gCAAYB,eAAzB,C;IAHsD,C;;MAwB3E,sBAAO,CAAP,C;QAD+E,OAC/D,sB
AAS,GAAT,MAAJ,GAakB,GAAIB,GAA2B,aAAM,mBAAiB,GAAjB,EAA8B,KAAtB,EAA6B,IAA7B,CAAN,C;
WACvC,sBAAO,CAAP,C;QAF+E,OAE/D,sBAAS,GAAT,MAAJ,GAakB,GAAIB,GAA2B,QAAM,mBAAiB,KA
AjB,EAAwB,GAAxB,EAA8B,IAAD,aAA7B,CAAN,C;;QAC/B,MAAA,gCAAYB,eAAzB,C;IAH0D,C;;IC
IDnF,C;;MAEgC,sB;IAAA,C;;IAMhC,C;;MAEgC,mC;IAAA,C;;IAMhC,C;;MAEgC,uB;IAAA,C;;IAMhC,C;;M
AEgC,qB;IAAA,C;;IAMhC,C;;MAEgC,sB;IAAA,C;;IAMhC,C;;MAEgC,uB;IAAA,C;;IAMhC,C;;MAEgC,wB;I
AAA,C;;IAMhC,C;;MAEgC,yB;IAAA,C;;MCDiD,uB;MAAjB,gB;MAC5D,sBAA2B,IAAK,I;MACHc,iBAAMC

,YAAO,CAAX,GAAC,SAAS,IAAvB,GAAiC,SAAS,I;MACzE,cAAuB,cAAJ,GAAa,KAAM,IAAnB,GAAgC,mB;IAHjB,C;;MAKA,qB;IAAA,C;;MAG9B,YAAY,W;MACZ,IAAI,UAAS,mBAAb,C;QACI,IAAI,CAAC,cAAL,C;UAAc,MAAa,6B;QAC3B,iBAAU,K;;QAGV,4BAAQ,SAAR,I;;MAEJ,OAAa,OAAN,KAAM,C;IACjB,C;;MAO0E,sB;MAAJB,gB;MACzD,sBAA2B,I;MAC3B,iBAAmC,YAAO,CAAX,GAAC,SAAS,IAAvB,GAAiC,SAAS,I;MACzE,cAAuB,cAAJ,GAAa,KAAb,GAAwB,mB;IAHV,C;;MAKC,qB;IAAA,C;;MAG9B,YAAY,W;MACZ,IAAI,UAAS,mBAAb,C;QACI,IAAI,CAAC,cAAL,C;UAAc,MAAa,6B;QAC3B,iBAAU,K;;QAGV,4BAAQ,SAAR,I;;MAEJ,OAAO,K;IACX,C;;MAO8E,uB;MAAIB,gB;MAC5D,sBAA2B,I;MAC3B,iBAAmC,uBAAO,CAAX,GAAC,sBAS,IAAT,MAAd,GAAiC,sBAAS,IAAT,M;MACHe,cAAuB,cAAJ,GAAa,KAAb,GAAwB,mB;IAHT,C;;MAKA,qB;IAAA,C;;MAG9B,YAAY,W;MACZ,IAAI,cAAS,mBAAT,CAAJ,C;QACI,IAAI,CAAC,cAAL,C;UAAc,MAAa,6B;QAC3B,iBAAU,K;;QAGV,8BAAQ,SAAR,C;;MAEJ,OAAO,K;IACX,C;;MCvBA,uC;MAjCI,IAAI,SAAQ,CAAZ,C;QAAe,MAAa,gCAAYB,wBAAzB,C;MAC5B,IAAI,SAAQ,WAAZ,C;QAA2B,MAAa,gCAAYB,wEAAzB,C;MAG5C,aAGyB,K;MAEzB,YAG6F,OAArE,0BAA0B,KAAM,IAAhC,EAAyC,YAAa,IAAtD,EAA+D,IAA/D,CAAqE,C;MAE7F,YAGuB,I;IAxBvB,C;;MA0BwC,mCAAwB,UAAxB,EAA+B,SAA/B,EAAqC,SAArC,C;IAAA,C;;MAGH,OAAI,YAAO,CAAX,GAAC,aAAQ,SAAtB,GAAgC,aAAQ,S;IAAx,C;;MAGjC,iDAA6B,kBAAa,KAAAM,UAAnB,KAC7B,eAAS,KAAM,MAAf,IAAwB,cAAQ,KAAM,KAAtC,IAA8C,cAAQ,KAAM,KAD/B,CAA7B,C;IAAA,C;;MAIA,OAAI,cAAJ,GAAe,EAAf,GAAwB,OAAM,OAAK,UAAM,IAAX,UAAqB,SAAK,IAA1B,KAAAN,SAA2C,SAA3C,I;IAAx,C;;MAE8B,OAAI,YAAO,CAAX,GAAC,oBAAE,UAAF,+BAAU,SAAV,eAAqB,SAAnc,GAA8C,oBAAE,UAAF,qCAAgB,SAAhB,gBAA4B,CAAC,SAAD,IAA5B,C;IAA9C,C;;MAEIC,yC;IAAA,C;;MAS+F,2BAAgB,UAAhB,EAA4B,QAA5B,EAA5C,IAAtC,C;IAAA,C;;MAT/F,gD;QAAA,+B;;MAAA,yC;IAAA,C;;MAYDA,sC;MAjCI,IAAI,SAAQ,CAAZ,C;QAAe,MAAa,gCAAYB,wBAAzB,C;MAC5B,IAAI,SAAQ,WAAZ,C;QAA2B,MAAa,gCAAYB,wEAAzB,C;MAG5C,aAGwB,K;MAExB,YAG4F,0BAArC,KAAqC,EAAf,YAAe,EAAAN,IAAM,C;MAE5F,YAGuB,I;IAxBvB,C;;MA0BuC,kCAAuB,UAAvB,EAA8B,SAA9B,EAAoC,SAApC,C;IAAA,C;;MAGF,OAAI,YAAO,CAAX,GAAC,aAAQ,SAAtB,GAAgC,aAAQ,S;IAAx,C;;MAGjC,gDAA4B,kBAAa,KAAM,UAAnB,KAC5B,eAAS,KAAM,MAAf,IAAwB,cAAQ,KAAM,KAAtC,IAA8C,cAAQ,KAAM,KADhC,CAA5B,C;IAAA,C;;MAIA,OAAI,cAAJ,GAAe,EAAf,GAAwB,OAAM,MAAK,UAAAL,QAAa,SAAb,IAAN,SAA2B,SAA3B,I;IAAx,C;;MAE8B,OAAI,YAAO,CAAX,GAAGB,UAAF,qBAAU,SAAV,cAAqB,SAAnc,GAAgD,UAAF,2BAAgB,SAAhB,eAA4B,CAAC,SAAD,IAA5B,C;IAA9C,C;;MAEIC,wC;IAAA,C;;MAS4F,0BAAe,UAAf,EAA2B,QAA3B,EAAqC,IAArC,C;IAAA,C;;MAT5F,+C;QAAA,8B;;MAAA,wC;IAAA,C;;MAYDA,uC;MAjCI,IAAI,gBAAJ,C;QAAGB,MAAa,gCAAYB,wBAAzB,C;MAC7B,IAAI,sCAAJ,C;QAA4B,MAAa,gCAAYB,yEAAzB,C;MAG7C,aAGyB,K;MAEzB,YAG+F,4BAAvC,KAAuC,EAAhB,YAAgB,EAAAN,IAAM,C;MAE/F,YAGwB,I;IAxBxB,C;;MA0BwC,mCAAwB,UAAxB,EAA+B,SAA/B,EAAqC,SAArC,C;IAAA,C;;MAGH,OAAI,uBAAO,CAAX,GAAC,2BAAQ,SAAR,KAAAd,GAAgC,2BAAQ,SAAR,K;IAAhC,C;;MAGjC,iDAA6B,kBAAa,KAAM,UAAnB,KAC7B,mBAAS,KAAM,MAAf,KAAb,kBAAQ,KAAM,KAAd,CAAxB,IAA8C,kBAAQ,KAAM,KAAd,CADjB,CAA7B,C;IAAA,C;;MAIA,OAAI,cAAJ,GAAe,EAAf,GAAwB,iCAAM,iCAAM,eAAW,8BAAW,EAAAX,CAAX,CAAN,MAAoC,cAAU,6BAAU,EAAV,CAAV,CAApC,CAAN,MAAuE,cAAU,6BAAU,EAAV,CAAV,CAAvE,CAAIg,Q;IAAZH,C;;MAE8B,OAAI,uBAAO,CAAX,GAAGB,UAAF,qBAAU,SAAV,yBAAqB,SAArB,WAAAd,GAAgD,UAAF,2BAAgB,SAAhB,yBAA6B,SAAD,aAA5B,W;IAA9C,C;;MAEIC,yC;IAAA,C;;MASgG,2BAAgB,UAAhB,EAA4B,QAA5B,EAA5C,IAAtC,C;IAAA,C;;MATHg,gD;QAAA,+B;;MAAA,yC;IAAA,C;;MChJkD,+BAAS,UAAT,UAAkB,wBAAS,iBAAT,M;IAAIB,C;;MAKIB,oCAAQ,iBAAR,K;IAAA,C;;M3CDhC,iC;MAjBsD,2BAAgB,KAAhB,EAAuB,YAAvB,EAAqC,CAArC,C;IAApC,C;;MACe,8B;IAAA,C;;MACO,6B;IAAA,C;;MAEM,qBAAS,KAAT,IAAkB,SAAS,S;IAA3B,C;;MAEZ,oBAAQ,S;IAAR,C;;MAG9B,2CAAuB,kBAAa,KAAM,UAAnB,KACvB,eAAS,KAAM,MAAf,IAAwB,cAAQ,KAAM,KADf,CAAvB,C;IAAA,C;;MAIA,OAAI,cAAJ,GAAe,EAAf,GAAwB,OAAM,IAAX,UAAqB,SAAK,IAA1B,K;IAAx,C;;MAE8B,2BAAE,UAAF,+BAAU,SAAV,C;IAAA,C;;MAEIC,mC;MACI,aAC8B,cAAY,OAAF,CAAe,CAAZ,EAAwB,OAAF,CAAe,CAAxB,C;IAFIC,C;;MAAA,0C;QAAA,yB;;MAAA,mC;IAAA,C;;MA0BA,gC;MAjBmD,0BAAe,KAAf,EAA5B,YAAtB,EAAoC,CAApC,C;IAAIC,C;;MACe,iB;IAAA,C;;MACO,gB;IAAA,C;;MAEM,qBAAS,KAAT,IAAkB,SAAS,S;IAA3B,C;;MAEX,oBAAQ,S;IAAR,C;;MAG9B,0CAAsB,kBAAa,KAAM,UAAnB,KACtB,eAAS,KAAM,MAAf,IAAwB,cAAQ,KAAM,KADhB,CAAtB,C;IAAA,C;;MAIA,OAAI,cAAJ,GAAe,EAAf,GAAwB,MAAK,UAAAL,QAAa,SAAb,I;IAAx,C;;

MAE8B,OAAE,UAAF,qBAAU,S;IAAV,C;;MAEIC,kC;MACI,aAC6B,aAAS,CAAT,EAAY,CAAZ,C;IAFjC,C;;;
MAAA,yC;QAAA,wB;;MAAA,kC;IAAA,C;;;MA0BA,iC;MAjBsD,2BAAgB,KAAhB,EAaUB,YAAvB,K;IAApC,
C;;MACe,iB;IAAA,C;;MACO,gB;IAAA,C;;MAEM,kCAAS,KAAT,UAAkB,sBAAS,SAAT,M;IAAIB,C;;MAEZ,k
CAAQ,SAAR,K;IAAA,C;;MAG9B,2CAAuB,kBAAa,KAAM,UAAuB,KACvB,mBAAS,KAAM,MAAf,KAAwB,k
BAAQ,KAAM,KAAd,CADD,CAAvB,C;IAAA,C;;MAIA,OAAI,CAAJ,GAAe,EAaf,GAAwB,iCAAM,eAAW,8B
AAW,EAAX,CAAX,CAAN,MAAoC,cAAU,6BAAU,EAaV,CAAU,CAApC,CAA8D,Q;IAAtF,C;;MAE8B,OAAE
,UAAF,qBAAU,SAaV,W;IAAA,C;;MAEIC,mC;MACI,aAC8B,qB;IAFIC,C;;;MAAA,0C;QAAA,yB;;MAAA,mC
;IAAA,C;;;M4CjCA,e;MAAA,iB;MAAA,uB;IAAA,C;;MAAA,yC;MAAA,C;MAKI,sE;MAGA,0F;M
AGA,gE;IAXJ,C;;;MAKI,4B;MAAA,wC;IAAA,C;;MAGA,4B;MAAA,kD;IAAA,C;;MAGA,4B;MAAA,qC;IAA
A,C;;MAXJ,sI;IAAA,C;;MAAA,a;AAAA,U;UAAA,6C;aAAA,oB;UAAA,uD;aAAA,O;UAAA,0C;gBAAA,mE;;I
AAA,C;;;MCwBA,uC;MATI,wB;MAIA,gB;IAL0B,C;;MAU9B,yC;MACI,Y
AMmC,oBAAgB,IAAhB,EAAsB,IAAtB,C;IAPvC,C;;MAeY,8DAAqC,IAArC,C;IAAA,C;;MAOA,uDAA8B,IAA9
B,C;IAAA,C;;MAOA,wDAA+B,IAA/B,C;IAAA,C;;;MA7BZ,gD;QAAA,+B;;MAAA,yC;IAAA,C;;MATI,oB;IA
XR,C;;MAeQ,gB;IAfR,C;;MAAA,2BAWQ,8CAXR,EAeQ,kCAfR,C;IAAA,C;;MAAA,0AWQ,4DAXR,IAeQ,sC
AfR,O;IAAA,C;;MAAA,c;MAWQ,yD;MAIA,qD;MAfR,a;IAAA,C;;MAAA,4IAWQ,4CAXR,IAeQ,oCAfR,I;IAA
A,C;;;MCICA,e;MAAA,iB;MAAA,uB;IAAA,C;;MAAA,mC;MAAA,C;MAYI,4D;MAKA,8C;MAKA,gD;IAtBJ,
C;;MAYI,sB;MAAA,mC;IAAA,C;;MAKA,sB;MAAA,4B;IAAA,C;;MAKA,sB;MAAA,6B;IAAA,C;;MAtBJ,m
G;IAAA,C;;MAAA,a;AAAA,W;UAAA,wC;aAAA,I;UAAA,iC;aAAA,K;UAAA,kC;gBAAA,6D;;IAAA,C;;MCA
A,e;MAAA,iB;MAAA,uB;IAAA,C;;MAAA,qC;MAAA,C;MAYI,0D;MAKA,gE;MAKA,8D;MAKA,4D;IA3BJ,C;;
;MAYI,wB;MAAA,kC;IAAA,C;;MAKA,wB;MAAA,qC;IAAA,C;;MAKA,wB;MAAA,oC;IAAA,C;;MAKA,wB;
MAAA,mC;IAAA,C;;MA3BJ,qJ;IAAA,C;;MAAA,a;AAAA,Q;UAAA,uC;aAAA,W;UAAA,0C;aAAA,U;UAAA,y
C;aAAA,S;UAAA,wC;gBAAA,+D;;IAAA,C;;MCAA,oB;IAAA,C;;MAI8B,oB;IAAA,C;;;MAJ9B,2B;QAAA,U;;
MAAA,oB;IAAA,C;;MCTQ,8B;MACI,aAAY,C;IADhB,C;;MAE6B,oBAAQ,gBAaI,O;IAAZ,C;;MACuB,Q;MAA
IB,IAAI,aAAQ,gBAaI,OAAhB,C;QAAA,OAAsB,iBAaI,iBAaJ,EAaI,yBAaJ,O;;;QAaKB,MAAM,2BAaYB,U
AAF,WAAvB,C;IAA9C,C;;;MAL9B,IAD8D,IAC9D,S;QACI,UAA0B,K;QAF0B,2C;;;QAAA,QAAM,IAAN,C;eA
SxD,c;YATwD,OAStC,qBAaQb,KAArB,C;eACIB,W;YAVwD,OAuZC,kBAaKB,KAAIB,C;eACf,Y;YAXwD,OA
WxC,mBAaMB,KAAuB,C;eAchB,W;YAZwD,OAYzC,kBAaKB,KAAIB,C;eACf,U;YAbwD,OAa1C,iBAaIB,K
AAjB,C;eACd,W;YAdwD,OAczC,kBAaKB,KAAIB,C;eACf,Y;YAfWd,OAexC,mBAaMB,KAAuB,C;eAchB,a;Y
AhBwD,OAgvC,oBAAoB,KAApB,C;kBACT,MAAM,6BAAsB,2DAA+C,IAA/C,CAAtB,C;;IAjBOC,C;;MAqBH
,kC;MAAS,0B;MAC9D,aAAY,C;IADyC,C;;MAE5B,oBAAQ,kBAAM,O;IAAd,C;;MACKC,Q;MAA9B,IAAI,aAA
Q,kBAAM,OAAIB,C;QAAA,OAAwB,mBAAM,iBAAN,EAAM,yBAAN,O;;;QAAoB,MAAM,2BAaYB,UAAF,W
AAvB,C;IAAID,C;;MAHwB,oD;IAAA,C;;MAON,kC;MAAS,uB;MACxD,aAAY,C;IADmC,C;;MAEtB,oBAAQ,
kBAAM,O;IAAd,C;;MAC+B,Q;MAA9B,IAAI,aAAQ,kBAAM,OAAIB,C;QAAA,OAAwB,mBAAM,iBAAN,EA
AM,yBAAN,O;;;QAAoB,MAAM,2BAaYB,UAAF,WAAvB,C;IAAID,C;;MAHqB,iD;IAAA,C;;MAOE,kC;MAAS,
wB;MAC1D,aAAY,C;IADqC,C;;MAExB,oBAAQ,kBAAM,O;IAAd,C;;MACgC,Q;MAA9B,IAAI,aAAQ,kBAAM,
OAAIB,C;QAAA,OAAwB,mBAAM,iBAAN,EAAM,yBAAN,O;;;QAAoB,MAAM,2BAaYB,UAAF,WAAvB,C;I
AAID,C;;MAHsB,kD;IAAA,C;;MAOF,kC;MAAS,uB;MACxD,aAAY,C;IADmC,C;;MAEtB,oBAAQ,kBAAM,O;
IAAd,C;;MAC+B,Q;MAA9B,IAAI,aAAQ,kBAAM,OAAIB,C;QAAA,OAAwB,mBAAM,iBAAN,EAAM,yBAAN,
O;;;QAAoB,MAAM,2BAaYB,UAAF,WAAvB,C;IAAID,C;;MAHqB,iD;IAAA,C;;MAOF,kC;MAAS,sB;MACtD,
aAAY,C;IADiC,C;;MAEpB,oBAAQ,kBAAM,O;IAAd,C;;MAC8B,Q;MAA9B,IAAI,aAAQ,kBAAM,OAAIB,C;Q
AAA,OAAwB,mBAAM,iBAAN,EAAM,yBAAN,O;;;QAAoB,MAAM,2BAaYB,UAAF,WAAvB,C;IAAID,C;;M
AHoB,gD;IAAA,C;;MAOI,kC;MAAS,wB;MAC1D,aAAY,C;IADqC,C;;MAExB,oBAAQ,kBAAM,O;IAAd,C;;M
ACgC,Q;MAA9B,IAAI,aAAQ,kBAAM,OAAIB,C;QAAA,OAAwB,mBAAM,iBAAN,EAAM,yBAAN,O;;;QAAo
B,MAAM,2BAaYB,UAAF,WAAvB,C;IAAID,C;;MAHsB,kD;IAAA,C;;MAOE,kC;MAAS,yB;MAC5D,aAAY,C;
IADuC,C;;MAE1B,oBAAQ,kBAAM,O;IAAd,C;;MACiC,Q;MAA9B,IAAI,aAAQ,kBAAM,OAAIB,C;QAAA,OA
AwB,mBAAM,iBAAN,EAAM,yBAAN,O;;;QAAoB,MAAM,2BAaYB,UAAF,WAAvB,C;IAAID,C;;MAHuB,mD
;IAAA,C;;MAOJ,kC;MAAS,uB;MACxD,aAAY,C;IADmC,C;;MAEtB,oBAAQ,kBAAM,O;IAAd,C;;MAC+B,Q;M
AA9B,IAAI,aAAQ,kBAAM,OAAIB,C;QAAA,OAAwB,mBAAM,iBAAN,EAAM,yBAAN,O;;;QAAoB,MAAM,2

BAAyB,UAAF,WAAvB,C;IAAID,C;;;MAHQb,iD;IAAA,C;;MAOnB,wB;IAAD,C;;;MAGe,MAAM,mC;IAAN,C;;
MAI1C,IAAI,qBAAJ,C;QACI,OAAO,CxB2GiF,WwB3GrE,UxB2GqE,EwB3GzD,QxB2GyD,C;;;QwBzGxF,OAA
S,CAAY,qBAAsB,UAAtB,EAAkC,QAAIC,C;;IAE7B,C;;MAII,IAAI,KAAy,kBAAhB,C;QACI,KAAy,mBAAkB,
QAAIB,EAA4C,sCAAhB,QAAgB,EAA5C,C;;;QAEH,QAAT,SAA+C,CAAlB,IAAjC,KAAiC,EAAkB,O;;IAEvD,
C;;MAKwB,Q;MADpB,gBAAgB,IAAhB,KAAgB,E;MACI,IAAI,OCjGkB,ODiGT,OAAT,EAAqB,WAArB,CAAJ
,C;QACHB,OAAI,aAAJ,GAAMB,KAAM,WAAzB,GAAYC,I;;;QAEzC,c;;MAHJ,wB;MAKA,kBAAkB,K;MACIB,
iBAAiB,W;MACjB,OAAO,S;IACX,C;;MAGyB,U;IAAD,C;;MAEHb,OAAO,mCAAsB,WAAK,KAAM,E;IAC5C,
C;;MAGI,OAAO,M;IACX,C;;MAGI,OAAuC,oBAAnB,UAA5B,IAAe,EAAa,CAAmB,C;IAC3C,C;;MAGI,OAAR
,IAAI,EAAW,GAAN,K;IACL,C;;MAII,OAAO,M;IACX,C;;;MAKA,YAAy,MAAY,IAAK,OAAjB,C;MACZ,sBA
AU,IAAV,a;QACI,UAAU,KAAK,CAAL,C;QACV,IAAI,oBAAJ,C;UACI,MAAM,CAAN,IAAW,EAAS,MAAM,
MAAK,GAAL,C;;;UAE1B,MAAM,CAAN,IAAW,G;;;MAGnB,OAAO,EAAS,OAAO,OAAM,EAAN,EAAGB,KA
AhB,C;IAC3B,C;;MAQW,WAAO,S;MAIBd,YAAy,MAAY,IAAK,OAAjB,C;MACZ,sBAAU,IAAV,a;QACI,UA
AU,KAAK,CAAL,C;QACV,IAAI,oBAAJ,C;UACI,MAAM,CAAN,IAAW,EAAS,MAAM,MAAK,GAAL,C;;;UA
E1B,MAAM,CAAN,IAAW,G;;;MAYnB,OATO,EAAS,OAAO,OAAM,EAAN,EAAGB,KAHhB,C;IAU3B,C;;MAA
I,WAAqB,S;MACrB,IAAI,qBAAMb,CAAY,OAAd,KAA2B,SAAhD,C;QAJCA,YAAy,MAkCM,IAICW,OAAjB,
C;QACZ,sBAiCkB,IAjClB,a;UACI,UAGc,IahCJ,CAAK,CAAL,C;UACV,IAAI,oBAAJ,C;YACI,MAAM,CAAN,
IAAW,EAAS,MAAM,MAAK,GAAL,C;;;YAE1B,MAAM,CAAN,IAAW,G;;;QA4Bf,OAzBG,EAAS,OAAO,OA
M,EAAN,EAAGB,KAHhB,C;;;QA2BnB,WAAW,C;QACX,0BAAU,IAAV,e;UACY,IAAoB,I;UAA5B,eAAQ,QA
AoB,OAAPb,IAAQ,CAAH,GAAG,CAAY,OAAPb,oCAAR,K;;QAEJ,aAAa,IAAjB,CAAC,YAAGB,CAAH,IAAG
,C;QEzFjB,IF0FyB,CE1FhB,OAAL,KAAkB,SAAtB,C;UF0F4B,MEzFxB,UFyFqB,CEzFF,O;;QF0FnB,OAAO,C;
QACP,0BAAU,IAAV,e;UAE0B,YACX,M;UAFX,YAAU,IAAQ,CAAH,GAAG,C;UACI,SAAJ,KAAI,O;UAAtB,a
AAU,CAAV,kB;YACI,OAAO,aAAP,EAAO,qBAAP,YAAiB,MAAI,CAAJ,C;;;QAGzB,OAAO,M;;IAEf,C;;MAG
gC,WAAS,c;MAAT,YAAhC,EAAE,MAAM,KAAiD,CAA3C,SAA2C,C;MAWrD,eAAiB,I;MAXW,OAYrB,K;IA
ZqB,C;;MAGH,WAAS,W;MAAT,YAAsB,IAA/C,WAA+C,CAAnC,EAAE,MAAM,KAAK,CAAC,SAAD,CAAsB
,C;MAQ/C,eAAiB,I;MARQ,OASIB,K;IATkB,C;;MAGA,WAAS,W;MAAT,YAA7B,EAAE,MAAM,KAA2C,CAA
rC,SAAqC,C;MAK/C,eAAiB,I;MALQ,OAMIB,K;IANkB,C;;MAKzB,eAAiB,I;MACjB,OAAO,K;IACX,C;;MGz
MsC,8C;MACIC,eAAsB,C;MACTb,wBAA+B,C;MAC/B,gBAA6B,I;MAC7B,mBAAsC,I;MACTc,qBAAyC,I;MA
EzC,yBAAGd,yBAAMb,Q;MAEnE,sBAAgD,I;IATf,C;;MAOjC,6B;IAAA,C;;MAMY,kBADr,M;MAAA,U;MAA
A,2C;QAAA,e;;QAES,gBADD,2CAAQ,yCAAR,gDAAwD,IAAxD,6BAAiE,I;QACzD,sBxCwEd,S;QwC1EF,SxC
2EG,S;;MwC3EH,a;IAAA,C;;MAKA,kBAAc,IAAd,C;MACiC,oB;MCqBrB,Q;MADR,IDpBsB,MCObTb,W;QADJ
,mBACiB,I;;QADjB,mBAEY,QDrBc,MCqBd,+D;;MDrBZ,yC;MACA,2BAAMc,MAAO,kBAA1C,C;MAGA,OA
AO,IAAP,C;QxCoCY,gBwCnCH,S;;QACD,iBAAiB,8B;QAGjB,IAAI,0BAAJ,C;UACI,qBAAc,e;;UAEd,oBAAQ,
0B;UACR,wBAAy,kB;;UAIZ,cAAc,oB;UACd,IAAI,YAAy,yBAAhB,C;YAAqC,M;UACrC,kBAAgB,O;UACHb
,qBAAMb,I;;UAEnB,kBAAGb,I;UACHb,qBAAMb,S;;QAGvB,gC;QAEA,IAAI,wCAAJ,C;UAEI,YAAU,U;;UA
GV,U;UAAA,0C;YEVhB,8BD2CQ,WAAO,qBAAP,CC3CR,C;YFUGb,a;;YAAA,a;UAAA,mB;YAEK,UErBrB,o
BD6CQ,WDXb+B,eCwB/B,CC7CR,C;;UFsBgB,M;;IAIhB,C;;MAGI,kBAAkB,mB;MACIB,IAAI,uBAAuB,gBA
AgB,IAA3C,C;QACI,uCAAQ,yCAAR,EAAMc,wCAA+B,WAA/B,C;;MAEvC,sBAAoB,mC;IACxB,C;;;MAKJ,q
C;IAAA,C;;M5C4DiD,MAAM,6B4C1DjC,uC5C0D+D,WAA9B,C;I4C1DvC,C;;M5C0DiC,MAAM,6B4CvDzC,u
C5CuDuE,WAA9B,C;I4CtDnD,C;;MAEkC,8C;IAAA,C;;;MARTc,4C;QAAA,2B;;MAAA,qC;IAAA,C;IGmDA,m
G;IAAA,yH;;MAKW,kC;MAAS,4C;IALpB,C;IAAA,kF;MAOY,Q;MAAA,sC;QAAiB,U;;MACjB,OAAO,oB;IAC
X,C;IATR,6G;;MA3GU,OAAK,SAAL,CAAiB,UAAjB,EAA6B,KAA7B,C;IAAA,C;;MAMBA,OAAK,SAAL,CA
AiB,QAAjB,EAA2B,UAA3B,EAAuC,KAAvC,C;IAAA,C;;MAiC6C,mB;QAC3C,OAAK,iCAAL,CAAiB,kBAAJB
,C;MAD2C,C;IAAA,C;;MAJnD,IAAS,SAAY,OAAjB,IAA2B,CAA/B,C;QAAA,OAES,SAAL,CAAiB,UAAjB,EA
A6B,IAA7B,C;;;QA8D0B,Q;QAhE9B,4DAImD,0DAJnD,EAGe8B,qBA5DS,UA4DT,qCAhE9B,C;;IAAA,C;;MA
wCmD,mB;QAC3C,OAAK,iCAAL,CAAiB,gBAAjB,EAA2B,kBAA3B,C;MAD2C,C;IAAA,C;;MAJnD,IAAS,SA
AY,OAAjB,IAA2B,CAA/B,C;QAAA,OAES,SAAL,CAAiB,QAAjB,EAA2B,UAA3B,EAAuC,IAAvC,C;;;QA0B0
B,Q;QA5B9B,4DAImD,sEAJnD,EA4B8B,qBAxBs,UAWBT,qCA5B9B,C;;IAAA,C;;MAoBC,kBAAD,M;MAAA,
kBAAC,qEAAD,4DAA2C,S;IAA3C,C;6CAGJ,yB;MAAA,mG;MAAA,yH;;QAKW,kC;QAAS,4C;MALpB,C;MA

AA,kF;QAOY,Q;QAAA,sC;UAAiB,U;;QACjB,OAAO,oB;MACX,C;MATR,6G;MAAA,oC;QAKkC,Q;QAA9B,m
EAA8B,oEAA9B,C;MAMJ,C;KAXA,C;;MCxH6C,OAAA,MAAa,YAAW,CAAX,C;IAAb,C;;MCDzC,OAAAG,GA
AG,CAAC,QAAD,C;IACV,C;;MAII,OAAJ,GAAL,GAAE,G;IAAF,C;;MPL0C,cAAvC,C;IAAuC,C;;MCAC,S;IAA
A,C;;MAgCa,0BAAe,WAAf,C;IAAA,C;;MAM4B,0BAAe,WAAf,C;IAAA,C;;MAOhB,0BAAe,WAAf,C;IAAA,C;
;MAOV,Q;MAH1D,aAAkB,GAAL,O;MACtB,aAAkB,GAAL,O;MACtB,YAAiB,C;MACjB,OAAO,QAAQ,MAAR,
IAAkB,QAAQ,MAAjC,C;QAAYC,IAAI,KAAJ,IAAa,IAAI,YAAJ,EAAI,oBAAJ,O;;MACtD,OAAO,G;IACX,C;;
MASuC,Q;MALnC,aAAa,MAAO,OAAAM,CAAN,EAAS,OAAT,C;MA0BpB,IAzBc,MAyBL,OAAL,KAAkB,SAA
tB,C;QAzBsB,MA0BIB,UAlBU,MA0BS,O;;MAzBvB,YAAiB,MAAO,O;MACxB,IAAI,UAAU,KAAAd,C;QACI,g
BAAgB,O;QACHB,OAAO,QAAQ,OAAf,C;UAAwB,OAAO,YAAP,EAAO,oBAAP,UAAkB,Y;;;MAE9C,OAAO,
M;IACX,C;;MAOoB,UAAmB,M;MAJnC,aAAa,KAAM,Q;MACnB,MAAO,OAAP,IAAiB,UAAW,K;MAc5B,IAb
c,KAAaL,OAAL,KAAkB,SAAtB,C;QAbqB,MAcJB,UAdU,KAcS,O;;MAbvB,YAAiB,KAAM,O;MACP,4B;MAAh
B,OAAgB,cAAhB,C;QAAgB,yB;QAAY,OAAO,cAAP,EAAO,sBAAP,YAAkB,O;;MAC9C,OAAO,M;IACX,C;;M
AlOB,UAAgB,M;MADhC,YAAy,U;MACI,4B;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QAAY,IAAI,cAAJ,EAAI,
sBAAJ,YAAe,O;;MAC3C,OAAO,G;IACX,C;;MAGI,IAAI,IAAK,OAAL,KAAkB,SAAtB,C;QACI,YAAc,IAAK,
O;;IAE3B,C;;MAE+D,OAAA,MAAa,QAAO,GAAP,EAAY,OAAZ,C;IAAb,C;;MOxF3D,kBAAkB,MAAa,eAAc,S
AAAd,C;MAC/B,iBAAiB,MAAa,eAAc,IAAd,C;MAC9B,OAAW,gBAAe,UAAAnB,GAA+B,SAA/B,GAAyC,CAAC,
S;IACrD,C;;MCcyE,OAAA,MAAa,gBAAe,IAAf,C;IAAb,C;;MA8BH,OAAA,MAAa,eAAc,IAAd,C;IAAb,C;;MA
GtB,OAAA,MAAa,KAAK,UAAAS,GAAT,EAAc,IAAd,C;IAAiB,C;;MACN,OAAK,SAAY,a;IAAjB,C;;MACC,OA
AK,SAAY,c;IAAjB,C;;MCpDW,OAAA,SAAY,UAAAS,WAAW,KAAX,CAAT,C;IAAZ,C;;MjEmf3C,Q;MAAA,Ic
xegB,KdweZ,IAAS,CAAT,IcxeY,KdweE,IAAS,wBAA3B,C;QAAA,OAAsC,UcxeT,KdwesB,C;;Qcxeb,MAAM,8
BAA0B,iCAAuB,gBAAvB,MAA1B,C;;MAAtC,W;IACJ,C;;MdifW,Q;MAAA,IczegB,KdyeZ,IAAS,CAAT,IcyeY,
KdyeE,IAAS,0BAA3B,C;QAAA,OAAsC,UcxeT,KdyesB,C;;Qczeb,MAAM,8BAA0B,iCAAuB,gBAAvB,MAA1B
,C;;MAAtC,W;IACJ,C;;MdkfW,Q;MAAA,Ic1egB,Kd0eZ,IAAS,CAAT,Ic1eY,Kd0eE,IAAS,0BAA3B,C;QAAA,O
AAsC,Uc1etB,Kd0esB,C;;Qc1eb,MAAM,8BAA0B,iCAAuB,gBAAvB,MAA1B,C;;MAAtC,W;IACJ,C;;MdmfW,Q
;MAAA,Ic3egB,Kd2eZ,IAAS,CAAT,Ic3eY,Kd2eE,IAAS,0BAA3B,C;QAAA,OAAsC,Uc3etB,Kd2esB,C;;Qc3eb,
MAAM,8BAA0B,iCAAuB,gBAAvB,MAA1B,C;;MAAtC,W;IACJ,C;;MdofW,Q;MAAA,Ic5egB,Kd4eZ,IAAS,CA
AT,Ic5eY,Kd4eE,IAAS,0BAA3B,C;QAAA,OAAsC,Uc5etB,Kd4esB,C;;Qc5eb,MAAM,8BAA0B,iCAAuB,gBAAv
B,MAA1B,C;;MAAtC,W;IACJ,C;;MdqfW,Q;MAAA,Ic7egB,Kd6eZ,IAAS,CAAT,Ic7eY,Kd6eE,IAAS,0BAA3B,C
;QAAA,OAAsC,Uc7etB,Kd6esB,C;;Qc7eb,MAAM,8BAA0B,iCAAuB,gBAAvB,MAA1B,C;;MAAtC,W;IACJ,C;;
MdsfW,Q;MAAA,Ic9egB,Kd8eZ,IAAS,CAAT,Ic9eY,Kd8eE,IAAS,0BAA3B,C;QAAA,OAAsC,Uc9etB,Kd8esB,C
;;Qc9eb,MAAM,8BAA0B,iCAAuB,gBAAvB,MAA1B,C;;MAAtC,W;IACJ,C;;MdufW,Q;MAAA,Ic/egB,Kd+eZ,I
AAS,CAAT,Ic/eY,Kd+eE,IAAS,0BAA3B,C;QAAA,OAAsC,Uc/etB,Kd+esB,C;;Qc/eb,MAAM,8BAA0B,iCAAuB,
gBAAvB,MAA1B,C;;MAAtC,W;IACJ,C;;MdwfW,Q;MAAA,IchfgB,Kd6fZ,IAAS,CAAT,IchfY,Kd6fE,IAAS,0BA
A3B,C;QAAA,OAAsC,UchftB,Kd6fsB,C;;Qchfb,MAAM,8BAA0B,iCAAuB,gBAAvB,MAA1B,C;;MAAtC,W;IA
CJ,C;;MAMI,OAAO,cAAa,SAAb,C;IACX,C;oFAEA,yB;MAAA,gD;MAAA,4B;QAKI,OAAsC,OAA/B,SAA+B,C
;MAC1C,C;KANA,C;oFAQA,yB;MAAA,gD;MAAA,4B;QAKI,OAAsC,OAA/B,SAA+B,C;MAC3C,C;KANA,C;
oFAQA,yB;MAAA,gD;MAAA,4B;QAKI,OAAsC,OAA/B,SAA+B,C;MACzC,C;KANA,C;oFAQA,yB;MAAA,g
D;MAAA,4B;QAKI,OAAsC,OAA/B,SAA+B,C;MAC1C,C;KANA,C;oFAQA,yB;MAAA,gD;MAAA,4B;QAKI,O
AAuC,OAAsC,SAAG,C;MAC3C,C;KANA,C;oFAQA,yB;MAAA,gD;MAAA,4B;QAKI,OAAsC,OAA/B,SAAi
C,C;MAC5C,C;KANA,C;oFAQA,yB;MAAA,gD;MAAA,4B;QAKI,OAAsC,OAAsC,SAAG,C;MAC7C,C;KANA
,C;;MAYW,8B;MAAS,uB;IAAT,C;;MAC4B,OAAA,gBAAy,O;IAAZ,C;;MACG,OAAA,gBdmwM/B,YAAQ,C;Ic
nwMuB,C;;MACc,OAAY,WAAZ,gBAAy,EAAS,OAAT,C;IAAZ,C;;MAE5C,oCAAA,2BAAkB,KAAIB,EAAyB,S
AAzB,C;MACb,OAAO,6BAAy,KAAZ,E;IACX,C;;MAES,Q;MAAL,IAAI,eAAC,uFAAD,CAAJ,C;QAAgC,OAA
O,E;MACvC,OAAM,B,UAAZ,gBAAy,EAQ,OAAR,C;IACvB,C;;MAES,Q;MAAL,IAAI,eAAC,uFAAD,CAAJ,C
;QAAgC,OAAO,E;MACvC,OAAM,B,cAAZ,gBAAy,EAAY,OAAsC;IACvB,C;;MAfJ,0C;IAiBJ,C;wFAsVA,yB;
MAAA,8C;MAAA,kF;QAmB0E,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,
gB;QACvI,UAAU,SAAV,EAAGB,WAAhB,EAA6B,iBAA7B,EAAGD,UAAhD,EAA4D,QAA5D,C;QACA,OAAO
,W;MACX,C;KATBA,C;wFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAyB,C;QAAG,0B;UAAA,

aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA0C,WAA1C,EAAiF,iBAAjF,EAAoG,UAAp
G,EAAgH,QAaHh,C;QACA,OAAO,W;MACX,C;KatBA,C;wFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;U
AAA,oBAAYB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA2C,W
AA3C,EAAmF,iBAAnF,EAAgS,UAAtG,EAaKH,QAAIH,C;QACA,OAAO,W;MACX,C;KatBA,C;wFAwBA,yB;
MAAA,8C;MAAA,kF;QAmBkE,iC;UAAA,oBAAYB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,
gB;QAC/H,UAAU,SAAV,EAAyC,WAAzC,EAA+e,iBAAE,EAAkG,UAAIG,EAA8G,QAA9G,C;QACA,OAAO,
W;MACX,C;KatBA,C;wFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAYB,C;QAAG,0B;UAAA,a
AAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA0C,WAA1C,EAAiF,iBAAjF,EAAoG,UAAp
G,EAAgH,QAaHh,C;QACA,OAAO,W;MACX,C;KatBA,C;wFAwBA,yB;MAAA,8C;MAAA,kF;QAmBsE,iC;U
AAA,oBAAYB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACnI,UAAU,SAAV,EAA2C,W
AA3C,EAAmF,iBAAnF,EAAgS,UAAtG,EAaKH,QAAIH,C;QACA,OAAO,W;MACX,C;KatBA,C;uFAwBA,yB;
MAAA,8C;MAAA,kF;QAmBwE,iC;UAAA,oBAAYB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB
,gB;QACrI,UAAU,SAAV,EAA4C,WAA5C,EAAqF,iBAArF,EAawG,UAAxG,EAAoH,QAApH,C;QACA,OAAO,
W;MACX,C;KatBA,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;UAAA,oBAAYB,C;QAAG,0B;UAAA,a
AAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACvI,UAAU,SAAV,EAA6C,WAA7C,EAAuF,iBAAvF,EAA0G,UAA
IG,EAAhS,QAAhT,C;QACA,OAAO,W;MACX,C;KatBA,C;yFAwBA,yB;MAAA,8C;MAAA,kF;QAmBoE,iC;U
AAA,oBAAYB,C;QAAG,0B;UAAA,aAAkB,C;QAAG,wB;UAAA,WAAgB,gB;QACjI,UAAU,SAAV,EAA0C,WA
A1C,EAAiF,iBAAjF,EAAoG,UAApG,EAAgH,QAaHh,C;QACA,OAAO,W;MACX,C;KatBA,C;;MA+BI,OAAAY
,SAAY,Q;IAC5B,C;;MASI,OAAAY,SAAY,Q;IAC5B,C;;MASI,OAAAY,SAAY,Q;IAC5B,C;;MASI,OAAAY,SAAY,Q
;IAC5B,C;;MAQW,WAAS,W;MAAT,YAA2B,SAAY,Q;MwCjkB9C,eAAiB,I;MxCikBjB,OwChkBO,K;IxCikBX,
C;;MASI,OAAAY,SAAY,Q;IAC5B,C;;MASI,OAAAY,SAAY,Q;IAC5B,C;;MAQW,WAAS,c;MAAT,YAA8B,SAAY
,Q;MwC9IBjD,eAAiB,I;MxC8IBjB,OwC7lBO,K;IxC8IBX,C;;MAQW,WAAS,W;MAAT,YAA2B,SAAY,Q;MwCv
mB9C,eAAiB,I;MxCumBjB,OwCtmBO,K;IxCumBX,C;;MDvxBI,IAAI,ECmyBI,WAAW,CDnyBf,CAAJ,C;QACI
,cCkyBoB,0C;QDjyBpB,MAAM,gCAAYB,OAAQ,WAAjC,C;;MCkyBV,OAAO,SAAS,SAAT,EAAe,cAAU,OAA
V,CAAf,C;IACX,C;;MDryBI,IAAI,ECizBI,WAAW,CDjzBf,CAAJ,C;QACI,cCgzBoB,0C;QD/yBpB,MAAM,gCA
AyB,OAAQ,WAAjC,C;;MCgzBV,OAAO,SAAS,SAAT,EAAe,eAAW,OAAAX,CAAf,C;IACX,C;;MDnzBI,IAAI,E
C+zBI,WAAW,CD/zBf,CAAJ,C;QACI,cC8zBoB,0C;QD7zBpB,MAAM,gCAAYB,OAAQ,WAAjC,C;;MC8zBV,O
AAO,SAAS,SAAT,EAAe,eAAS,OAAAT,CAAf,C;IACX,C;;MDj0BI,IAAI,EC60BI,WAAW,CD70Bf,CAAJ,C;QAC
I,cC40BoB,0C;QD30BpB,MAAM,gCAAYB,OAAQ,WAAjC,C;;MC40BH,WAAS,W;MAAT,YAAsB,gBAAGB,SA
AhB,EAAsB,OAAtB,K;MwC/pB7B,eAAiB,I;MxC+pBjB,OwC9pBO,K;IxC+pBX,C;;MD/0BI,IAAI,EC21BI,WAA
W,CD31Bf,CAAJ,C;QACI,cC01BoB,0C;QDz1BpB,MAAM,gCAAYB,OAAQ,WAAjC,C;;MC01BV,OAAO,SAAS
,SAAT,EAAe,iBAAW,OAAAX,CAAf,C;IACX,C;;MD71BI,IAAI,ECy2BI,WAAW,CDz2Bf,CAAJ,C;QACI,cCw2B
oB,0C;QDv2BpB,MAAM,gCAAYB,OAAQ,WAAjC,C;;MCw2BV,OAAO,SAAS,SAAT,EAAe,iBAAAY,OAAZ,CA
Af,C;IACX,C;;MD32BI,IAAI,ECu3BI,WAAW,CDv3Bf,CAAJ,C;QACI,cC3BoB,0C;QDr3BpB,MAAM,gCAAYB
,OAAQ,WAAjC,C;;MCs3BH,WAAS,c;MAAT,YAAyB,gBAAGB,SAAhB,EAAsB,OAAtB,EAA+B,KAA/B,C;Mw
CzsBhC,eAAiB,I;MxCysBjB,OwCxsBO,K;IxCysBX,C;;MDz3BI,IAAI,ECq4BI,WAAW,CDr4Bf,CAAJ,C;QACI,c
Co4BoB,0C;QDn4BpB,MAAM,gCAAYB,OAAQ,WAAjC,C;;MCo4BH,WAAS,W;MAAT,YAAsB,SAAS,SAAT,E
AAe,iBAAU,OAAV,CAAf,C;MwCvtB7B,eAAiB,I;MxCutBjB,OwCttBO,K;IxCutBX,C;;MDv4BI,IAAI,EC05BI,
WAAW,CDp5Bf,CAAJ,C;QACI,cCm5BoB,0C;QDI5BpB,MAAM,gCAAYB,OAAQ,WAAjC,C;;MCm5BV,OAAO
,gBAAGB,SAAhB,EAAsB,OAAtB,EAA+B,IAA/B,C;IACX,C;;MAUI,oCAAa,2BAaKB,SAIIB,EAA6B,OAA7B,
EAAcS,gBAAtC,C;MACb,OAAAY,SAAY,OAAM,SAAN,EAAiB,OAAjB,C;IAC5B,C;;MASI,oCAAa,2BAaKB,S
AAIB,EAA6B,OAA7B,EAAcS,gBAAtC,C;MACb,OAAAY,SAAY,OAAM,SAAN,EAAiB,OAAjB,C;IAC5B,C;;MA
SI,oCAAa,2BAaKB,SAIIB,EAA6B,OAA7B,EAAcS,gBAAtC,C;MACb,OAAAY,SAAY,OAAM,SAAN,EAAiB,O
AAjB,C;IAC5B,C;;MASI,oCAAa,2BAaKB,SAIIB,EAA6B,OAA7B,EAAcS,gBAAtC,C;MACb,OAAAY,SAAY,O
AAM,SAAN,EAAiB,OAAjB,C;IAC5B,C;;MASI,oCAAa,2BAaKB,SAIIB,EAA6B,OAA7B,EAAcS,gBAAtC,C;
MACN,WAAS,W;MAAT,YAA2B,SAAY,OAAM,SAAN,EAAiB,OAAjB,C;MwC9xB9C,eAAiB,I;MxC8xBjB,Ow
C7xBO,K;IxC8xBX,C;;MASI,oCAAa,2BAaKB,SAIIB,EAA6B,OAA7B,EAAcS,gBAAtC,C;MACb,OAAAY,SA
AY,OAAM,SAAN,EAAiB,OAAjB,C;IAC5B,C;;MASI,oCAAa,2BAaKB,SAIIB,EAA6B,OAA7B,EAAcS,gBAAtC,

C;MACb,OAAY,SAAY,OAAM,SAAN,EAAiB,OAAjB,C;IAC5B,C;;MASI,oCAAa,2BAAkB,SAAiB,EAA6B,OA
A7B,EAAc,gBAAtC,C;MACN,WAAS,c;MAAT,YAA8B,SAAY,OAAM,SAAN,EAAiB,OAAjB,C;MwC/zBjD,e
AAiB,I;MxC+zBjB,OwC9zBO,K;IxC+zBX,C;;MASI,oCAAa,2BAAkB,SAAiB,EAA6B,OAA7B,EAAc,gBAAtC,
C;MACN,WAAS,W;MAAT,YAA2B,SAAY,OAAM,SAAN,EAAiB,OAAjB,C;MwC10B9C,eAAiB,I;MxC00BjB,
OwCz0BO,K;IxC00BX,C;;MAagD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MAC/E,oCAAa,2BAAkB,
SAAiB,EAA6B,OAA7B,EAAc,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;IACr
B,C;;MAagD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MAC/E,oCAAa,2BAAkB,SAAiB,EAA6B,OAA
7B,EAAc,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;IACrB,C;;MAakD,yB;QA
AA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MACjF,oCAAa,2BAAkB,SAAiB,EAA6B,OAA7B,EAAc,gBAAtC,
C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;IACrB,C;;MAa8C,yB;QAAA,YAAiB,C;MAAG,
uB;QAAA,UAAe,gB;MAC7E,oCAAa,2BAAkB,SAAiB,EAA6B,OAA7B,EAAc,gBAAtC,C;MACR,SAAY,MAA
K,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;IACrB,C;;MAagD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;
MAC/E,oCAAa,2BAAkB,SAAiB,EAA6B,OAA7B,EAAc,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,
EAAyB,OAAzB,C;IACrB,C;;MAakD,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MACjF,oCAAa,2BAA
kB,SAAiB,EAA6B,OAA7B,EAAc,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;IA
CrB,C;;MAaod,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MACnF,oCAAa,2BAAkB,SAAiB,EAA6B,OA
A7B,EAAc,gBAAtC,C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;IACrB,C;;MAasD,yB;QA
AA,YAAiB,C;MAAG,uB;QAAA,UAAe,gB;MACrF,oCAAa,2BAAkB,SAAiB,EAA6B,OAA7B,EAAc,gBAAtC,
C;MACR,SAAY,MAAK,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;IACrB,C;;MAagD,yB;QAAA,YAAiB,C;MAAG
uB;QAAA,UAAe,gB;MAC/E,oCAAa,2BAAkB,SAAiB,EAA6B,OAA7B,EAAc,gBAAtC,C;MACR,SAAY,MAA
K,OAAL,EAAc,SAAd,EAAyB,OAAzB,C;IACrB,C;;MAOI,OAAY,SAAY,QAAO,CAAQ,OAAR,CAAP,C;IAC5B
,C;iFAEA,yB;MAwIA,iD;MAxIA,qC;QAKI,OAwIO,gCaxIK,eAAY,OAAZ,EAwIL,C;MAvIX,C;KANA,C;iFAQ
A,yB;MAwIA,iD;MAxIA,qC;QAKI,OAwIO,gCaxIK,gBAAa,OAAb,EAwIL,C;MAvIX,C;KANA,C;iFAQA,yB;M
AwIA,iD;MAxIA,qC;QAKI,OAwIO,gCaxIK,gBAAW,OAAX,EAwIL,C;MAvIX,C;KANA,C;iFAQA,yB;MAwIA,
iD;MAxIA,qC;QAKI,OAwIO,gCaxIK,mBAAy,OAAZ,CAwIL,C;MAvIX,C;KANA,C;iFAQA,yB;MAwIA,iD;M
AxIA,qC;QAKI,OAwIO,gCaxIK,kBAAa,OAAb,EAwIL,C;MAvIX,C;KANA,C;iFAQA,yB;MAwIA,iD;MAxIA,q
C;QAKI,OAwIO,gCaxIK,kBAAc,OAAd,EAwIL,C;MAvIX,C;KANA,C;iFAQA,yB;MAwIA,iD;MAxIA,qC;QAKI
,OAwIO,gCaxIK,sBAae,OAaf,CAwIL,C;MAvIX,C;KANA,C;iFAQA,yB;MAwIA,iD;MAxIA,qC;QAKI,OAwIO,
gCaxIK,mBAAy,OAAZ,CAwIL,C;MAvIX,C;KANA,C;;MAaI,OAao,oBAAoB,SAApB,EAA0B,QAA1B,C;IAC
X,C;;MAMI,OAao,mBAAwB,UAAL,SAAK,EAAO,mBAAO,QAAS,KAAhB,IAAP,CAAxB,EAAcD,SAAK,OA
A3D,EAAiE,QAAjE,C;IACX,C;;MAMI,OAao,mBAAwB,UAAL,SAAK,EAAO,mBAAO,QAAS,KAAhB,IAAP,
CAAxB,EAAcD,SAAK,OAA3D,EAAiE,QAAjE,C;IACX,C;;MAMI,OAao,mBAAwB,UAAL,SAAK,EAAO,mB
AAO,QAAS,KAAhB,IAAP,CAAxB,EAAcD,SAAK,OAA3D,EAAiE,QAAjE,C;IACX,C;;MAMI,OAao,oBAAoB,
SAApB,EAA0B,QAA1B,C;IACX,C;;MAMI,OAao,mBAAwB,UAAL,SAAK,EAAO,mBAAO,QAAS,KAAhB,IA
AP,CAAxB,EAAcD,SAAK,OAA3D,EAAiE,QAAjE,C;IACX,C;;MAMI,OAao,mBAAwB,UAAL,SAAK,EAAO,
mBAAO,QAAS,KAAhB,IAAP,CAAxB,EAAcD,SAAK,OAA3D,EAAiE,QAAjE,C;IACX,C;;MAMI,OAao,oBAA
oB,SAApB,EAA0B,QAA1B,C;IACX,C;;MAMI,OAao,mBAAwB,UAAL,SAAK,EAAO,mBAAO,QAAS,KAAhB
,IAAP,CAAxB,EAAcD,SAAK,OAA3D,EAAiE,QAAjE,C;IACX,C;;MAOI,OAAY,SAAY,QAAO,QAAP,C;IAC5B
,C;iFAEA,yB;MAAA,iD;MAAA,sC;QAKI,OAao,qBAAqB,SAArB,EAA2B,QAA3B,C;MACX,C;KANA,C;iFAQ
A,yB;MAAA,iD;MAAA,sC;QAKI,OAao,qBAAqB,SAArB,EAA2B,QAA3B,C;MACX,C;KANA,C;iFAQA,yB;M
AAA,iD;MAAA,sC;QAKI,OAao,qBAAqB,SAArB,EAA2B,QAA3B,C;MACX,C;KANA,C;iFAQA,yB;MAAA,iD
;MAAA,sC;QAKI,OAao,qBAAqB,SAArB,EAA2B,QAA3B,C;MACX,C;KANA,C;iFAQA,yB;MAAA,iD;MAAA
,sC;QAKI,OAao,qBAAqB,SAArB,EAA2B,QAA3B,C;MACX,C;KANA,C;iFAQA,yB;MAAA,iD;MAAA,sC;QA
KI,OAao,qBAAqB,SAArB,EAA2B,QAA3B,C;MACX,C;KANA,C;iFAQA,yB;MAAA,iD;MAAA,sC;QAKI,OA
AO,qBAAqB,SAArB,EAA2B,QAA3B,C;MACX,C;KANA,C;iFAQA,yB;MAAA,iD;MAAA,sC;QAKI,OA
AO,qBAAqB,SAArB,EAA2B,QAA3B,C;MACX,C;KANA,C;;MAaI,OAAY,SAAY,QAAO,CAAQ,OAAR,CAAP,C;IAC
5B,C;;MAk6C,OAaa,CAAE,iBAAU,CAAV,C;IAAa,C;;MAAxD,IAAI,mBAAO,CAAX,C;QAsGY,eAtGO,WA
sGP,C;;IARghB,C;;MA4DI,IAAI,mBAAO,CAAX,C;QAac,UAAU,SAAV,C;IACIB,C;;MAQI,IAAI,mBAAO,CA

CoPE,IA6+FT,gBAAR,iBAAQ,C;;MA7+FT,U;MAAA,S;QAAA,SAAsC,sBgCpPtB,KhCoPsB,C;;QgCpPb,MAAM ,8BAA0B,iCAAuB,cAAvB,MAA1B,C;;MAATc,a;IACJ,C;;MAQW,8B;MAAS,uB;IAAT,C;;MAC4B,OAAA,gBA AY,K;IAAZ,C;;MACG,OAAA,gBAAY,U;IAAZ,C;;MACc,OAAA,gBAAY,gBAAS,OAAT,C;IAAZ,C;;MAE5C,o CAAa,2BAAkB,KAAIB,EAAyB,SAAzB,C;MACb,OAAO,6BAAY,KAAZ,C;IACX,C;;MAES,Q;MAAL,IAAI,eA AC,0EAAD,OAAl,C;QAAgC,OAAO,E;MACvC,OhCmmBO,UgCnmBA,gBhCmmBR,QAAQ,EgCnmBoB,OjEyH F,KiC0elB,C;IgcImBX,C;;MAES,Q;MAAL,IAAI,eAAC,0EAAD,OAAl,C;QAAgC,OAAO,E;MACvC,OhC+zBO, cgC/zBA,gBhC+zBR,QAAQ,EgC/zBwB,OjEqHN,KiC0sB1B,C;Igc9zBX,C;;MAfJ,4C;IAiBJ,C;;MAQW,8B;MAA S,uB;IAAT,C;;MAC4B,OAAA,gBAAY,K;IAAZ,C;;MACG,OAAA,gBAAY,U;IAAZ,C;;MACe,OAAA,gBAAY,g BAAS,OAAT,C;IAAZ,C;;MAE7C,oCAAa,2BAAkB,KAAIB,EAAyB,SAAzB,C;MACb,OAAO,6BAAY,KAAZ,C; IACX,C;;MAES,Q;MAAL,IAAI,eAAC,0EAAD,QAAl,C;QAAiC,OAAO,E;MACxC,OhColBO,UgCplBA,gBhCol BR,QAAQ,EgCplBoB,OIC2GA,KEyepB,C;IgcnlBX,C;;MAES,Q;MAAL,IAAI,eAAC,0EAAD,QAAl,C;QAAiC,O AAO,E;MACxC,OhCgzBO,cgChzBA,gBhCgzBR,QAAQ,EgChzBwB,OICuGJ,KEysBpB,C;Igc/yBX,C;;MAfJ,4C ;IAiBJ,C;;MAQW,8B;MAAS,uB;IAAT,C;;MAC4B,OAAA,gBAAY,K;IAAZ,C;;MACG,OAAA,gBAAY,U;IAAZ, C;;MACe,OAAA,gBAAY,gBAAS,OAAT,C;IAAZ,C;;MAE7C,oCAAa,2BAAkB,KAAIB,EAAyB,SAAzB,C;MAC b,OAAO,6BAAY,KAAZ,C;IACX,C;;MAES,Q;MAAL,IAAI,eAAC,0EAAD,QAAl,C;QAAiC,OAAO,E;MACxC, OhCqkBO,UgCrkBA,gBhCqkBR,QAAQ,EgCrkBoB,OIE2CA,KkC0hBpB,C;IgcPkBX,C;;MAES,Q;MAAL,IAAI,e AAC,0EAAD,QAAl,C;QAAiC,OAAO,E;MACxC,OhCiyBO,cgCjyBA,gBhCiyBR,QAAQ,EgCjyBwB,OIEuCl,Kk C0vBpB,C;IgcHyBX,C;;MAfJ,4C;IAiBJ,C;;MAQW,8B;MAAS,uB;IAAT,C;;MAC4B,OAAA,gBAAY,K;IAAZ,C ;MACG,OAAA,gBAAY,U;IAAZ,C;;MACgB,OAAA,gBAAY,gBAAS,OAAT,C;IAAZ,C;;MAE9C,oCAAa,2BAA kB,KAAIB,EAAyB,SAAzB,C;MACb,OAAO,6BAAY,KAAZ,C;IACX,C;;MAES,Q;MAAL,IAAI,eAAC,0EAAD,S AAl,C;QAAkC,OAAO,E;MACzC,OhCsJBO,UgCtjBA,gBhCsJBR,QAAQ,EgCtjBoB,OhE6BE,KgCyhBtB,C;IgcRj BX,C;;MAES,Q;MAAL,IAAI,eAAC,0EAAD,SAAl,C;QAAkC,OAAO,E;MACzC,OhCkxBO,cgClxBA,gBhCkxBR,QAAQ,EgClxBwB,OhEyBF,KgCyvBtB,C;IgcJxBX,C;;MAfJ,4C;IAiBJ,C;;McrJ+C,OAAA,SAAY,Y;IAAZ,C;; MAGH,OAAA,SAAY,S;IAAZ,C;;MAGY,OAAA,SAAY,QAAO,MAAP,C;IAAZ,C;;MAGa,OAAA,SAAY,QAAO ,IAAP,EAAa,KAAb,C;IAAZ,C;;MAGb,OAAA,SAAY,QAAO,IAAP,C;IAAZ,C;;MAG4B,OAAA,SAAY,QAAO,I AAP,EAAa,KAAb,EAAoB,KAApB,C;IAAZ,C;;MAGxC,OAAA,SAAY,S;IAAZ,C;;MAGG,OAAA,SAAY,Y;IAA Z,C;;MAGY,OAAA,SAAY,WAAU,MAAV,C;IAAZ,C;;MAGX,OAAA,SAAY,a;IAAZ,C;;MczBjB,oB;QAAA,O AA0B,E;MAA1B,gB;IAAD,C;;IAE9B,C;;IAIA,C;;IAIA,C;;MAKkC,oB;QAAA,OAA0B,E;MAA1B,gB;IAAD, C;;IAEjC,C;;MAqCsC,gB;IAAD,C;;MAGCJ,sB;IAAD,C;;IAEhC,C;;MA8DoC,kB;IAAD,C;;IAEnC,C;;ICxJA, C;;IAIA,C;;MnDHqE,uCoCatB,E;IpcAsB,C;IGAerE,yB;MAAA,kD;MAAA,4B;QACsE,mBAAY,SAAZ,C;MA AA,C;KADtE,C;;MAMI,OAAsB,UAAy,QAAvB,KAAmC,SAa9C,GACe,UAAy,UAD3B,GAGI,gBAAgB,UAAh B,C;IACR,C;;MAII,YoChB2C,E;MpcB3C,eAAe,UAAW,W;MAC1B,OAAO,QAAS,UAAhB,C;QACU,KAAy,M AAK,QAAS,OAAd,C;MACtB,OAAO,K;IACX,C;;MAUc,Q;MANV,IAAI,KAAM,OAAN,GAAa,UAAW,KAA5B, C;QACI,OAAO,gBAAgB,UAAhB,C;;MAEX,eAAe,UAAW,W;MAC1B,YAAy,C;MACZ,OAAO,QAAS,UAAhB, C;QACI,MAAM,YAAN,EAAM,oBAAN,UAAiB,QAAS,O;;MAE9B,IAAI,QAAQ,KAAM,OAAIB,C;QACI,MAA M,KAAAN,IAAe,I;;MAEnB,OAAO,K;IACX,C;;MAK6C,sBAAY,OAAZ,E;IAAA,C;;MAKF,oBAAU,OAAV,E;IA AA,C;;MAMY,oBAAU,IAAV,E;IAAA,C;;MASIC,Q;MAAA,kC;MAAjB,iBAAc,CAAd,yB;QACI,sBAAK,KAAAL ,EAAc,KAAAd,C;;IAER,C;;MAQuD,qBAAQ,4BAAR,C;IAAA,C;;MAMiB,kBAAhB,0B;MAAwB,oB;MAAxB,OH R7C,W;IGQ6C,C;;MAQpD,gBAAgB,SAAhB,EAAsB,cAAtB,C;IACJ,C;;MAQI,gBAAgB,SAAhB,EAAsB,UAAtB ,C;IACJ,C;;MAGI,IAAI,IAAK,KAAAL,IAAa,CAAjB,C;QAAoB,M;MAEpB,YAAy,YAAy,IAAZ,C;MACZ,gBAA c,KAAAd,EAAqB,UAArB,C;MAEA,aAAU,CAAV,MAAkB,KAAM,OAAxB,M;QACI,iBAAK,CAAL,EAAU,MAA M,CAAN,CAAV,C;;IAER,C;;MAGI,OAAO,gBAAkB,IAAIB,O;IACX,C;;MAMI,oCAAa,2BAAkB,UAAIB,EAA8 B,QAA9B,EAAwC,MAAO,OAA/C,C;MACb,gBAAgB,WAAW,UAAx,I;MACHb,oCAAa,2BAAkB,iBAAIB,EA AqC,oBAAoB,SAApB,IAArC,EAAoE,WAAy,OAAhF,C;MAEb,IAAI,WAakB,QAAO,WAAP,CAAIB,IAAyC,W AAKB,QAAO,MAAP,CAA/D,C;QACI,eAAsB,MAAY,UAAU,UAAT,EAAqB,QAAR,C;QACtB,WAAy,KAAI,Q AAJ,EAAc,iBAAd,C;;QAExB,IAAI,WAAW,WAAx,IAA0B,qBAAqB,UAAAnD,C;UACI,iBAAc,CAAd,UAAsB,S AAtB,U;YACI,YAAy,oBAAoB,KAApB,IAAZ,IAAyC,OAAO,aAAa,KAAb,IAAP,C;;UAG7C,mBAAc,YAAy, CAAZ,IAAd,aAAmC,CAAnC,Y;YACI,YAAy,oBAAoB,OAAPB,IAAZ,IAAyC,OAAO,aAAa,OAAb,IAAP,C;;I

AIzD,C;;MAIgf,gB;IAAA,C;kGAehF,yB;MAAA,4D;MAAA,4B;QAC8E,OAAK,aAAL,SAAK,C;MAAL,C;KAD
9E,C;;MAMI,OAAI,SAAJ,GAEL,SAFJ,GAIL,SNwkBoB,Q;IM5kxB,C;;MAUA,IAAI,QAAQ,CAAZ,C;QACI,oB;;
MAEJ,OAAO,K;IACX,C;;MAII,IAAI,QAAQ,CAAZ,C;QACI,oB;;MAEJ,OAAO,K;IACX,C;;MoD1K2F,6B;IAAj
C,C;;MAKID,eAAe,e;MACf,OAAO,QAAS,UAAhB,C;QACI,IAAI,OAAA,QAAS,OAAI,EAAMb,OAAAnB,CAAJ
,C;UACI,QAAS,S;UACT,OAAO,I;;MAGf,OAAO,K;IACX,C;;MAIoB,Q;MADhB,eAAe,K;MACC,0B;MAAhB,O
AAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,eAAI,OAAJ,CAAJ,C;UAAkB,WAAW,I;;MAEjC,OAAO,Q;IACX,C;;
MAEyG,qB;QAAE,OAAO,gBAAN,mB;MAAF,C;IAAA,C;;MAAtC,Q;MAAD,OAA6B,YAA5B,iEAA4B,EAAU,
oDAAV,C;IAA7B,C;;MACuC,qB;QAAE,QAAO,gBAAP,mB;MAAF,C;IAAA,C;;MAAtC,Q;MAAD,OAA6B,YA
A5B,iEAA4B,EAAU,oDAAV,C;IAA7B,C;;MAG9D,eAAe,IAAK,W;MACpB,OAAO,QAAS,UAAhB,C;QACI,QA
AS,O;QACT,QAAS,S;;IAEjB,C;;MAGyB,OAAA,IAAK,U;IAAL,C;;MC9BwD,oC;MACjF,gBAA8B,C;IADkB,C;
;MAa5C,iBAAI,SAAJ,EAAU,OAAV,C;MACA,OAAO,I;IACX,C;;MAKc,UACF,M;MAHR,aAAa,K;MACb,cAAc
,K;MACJ,0B;MAAV,OAAU,cAAV,C;QAAU,mB;QACN,kBAAI,eAAJ,EAAI,uBAAJ,WAAc,CAAd,C;QACA,U
AAU,I;;MAEd,OAAO,O;IACX,C;;MAGI,yBAAY,CAAZ,EAAe,SAAf,C;IACJ,C;;MAE4E,qB;QAAE,OAAO,gB
AAN,mB;MAAF,C;IAAA,C;;MAAV,yBAAU,8CAAV,C;IAAA,C;;MACU,qB;QAAE,QAAO,gBAAP,mB;MAAF,
C;IAAA,C;;MAAV,yBAAU,8CAAV,C;IAAA,C;;MAGb,iD;IAAA,C;;MAED,0BAAQ,OAAAR,KAAoB,C;IAApB,
C;;MAG/B,Q;MAAA,6B;MAAJB,iBAAc,CAAd,yB;QACI,IAAI,wBAAI,KAAJ,GAAC,OAAAd,CAAJ,C;UACI,OA
AO,K;;MAGf,OAAO,E;IACX,C;;MAGI,iBAAc,sBAAd,WAA+B,CAA/B,U;QACI,IAAI,wBAAI,KAAJ,GAAC,O
AAd,CAAJ,C;UACI,OAAO,K;;MAGf,OAAO,E;IACX,C;;MAE6D,iCAAa,CAAb,C;IAAA,C;;MACU,sDAAiB,K
AAjB,C;IAAA,C;;MAGK,uCAAQ,IAAR,EAAc,SAAd,EAAyB,OAAzB,C;IAAA,C;;MAMxE,eAAe,0BAAa,SA
Ab,C;MACf,YAAO,UAAU,SAAV,I;MxDgEX,iBAAc,CAAd,UAAsB,KAAtB,U;QwD/DiB,e;QACA,iB;;IAEjB,C;;
MAQI,IAAI,UAAU,IAAd,C;QAAoB,OAAO,I;MAC3B,IAAI,2BAAJ,C;QAAuB,OAAO,K;MAE9B,OAAO,oCAA
a,uBAAc,IAAd,EAAoB,KAApB,C;IACxB,C;;MAK+B,OAAA,oCAAa,yBAAgB,IAAhB,C;IAAb,C;;MAG/B,oB;
MACI,eAcSb,C;MACtB,cAIqB,E;IAPzB,C;;MASsC,sBAAQ,gB;IAAR,C;;MAIvB,Q;MADP,IAAI,CAAC,cAAL,
C;QAAgB,MAAM,6B;MACtB,eAAO,mBAAP,EAAO,2BAAP,O;MACA,OAAO,wBAAI,WAAJ,C;IACX,C;;M5D
7BJ,IAAI,E4DgCU,gBAAQ,E5DhCIB,CAAJ,C;QACI,c4D+BwB,sE;Q5D9BxB,MAAM,6BAAsB,OAAQ,WAA9B
,C;;M4DgCF,6BAAS,WAAAT,C;MACA,eAAQ,W;MACR,cAAO,E;IACX,C;;MAGJ,oB;MAGmD,wD;MAG3C,oC
AAa,4BAAmB,KAAAnB,EAA0B,WAAyB,KAAAnD,C;MACb,eAAa,K;IAJe,C;;MAOM,sBAAQ,C;IAAR,C;;MAEN
,mB;IAAA,C;;MAG5B,IAAI,CAAC,kBAAL,C;QAAoB,MAAM,6B;MAE1B,eAAO,mCAAP,EAAO,YAAP,C;MA
CA,OAAO,wBAAI,WAAJ,C;IACX,C;;MAEoC,sBAAQ,CAAR,I;IAAA,C;;MAGhC,wBAAI,YAAJ,EAAW,OAA
X,C;MACA,mC;MACA,cAAO,E;IACX,C;;M5DnEJ,IAAI,E4DsEU,gBAAQ,E5DtEIB,CAAJ,C;QACI,c4DqEwB,4
E;Q5DpExB,MAAM,6BAAsB,OAAQ,WAA9B,C;;M4DqEF,wBAAYB,WAAzB,EAAiC,OAAjC,C;IACJ,C;;MAG
2G,8B;MAAtF,kB;MAA0C,4B;MAC/D,eAAyB,C;MAGrB,oCAAa,2BAAkB,gBAAlB,EAA6B,OAA7B,EAAc,
WAAK,KAA3C,C;MACb,eAAa,UAAU,gBAAV,I;IALG,C;;MASHb,oCAAa,4BAAmB,KAAAnB,EAA0B,YAA1B,
C;MAEb,WAAK,aAAI,mBAAY,KAAZ,IAAJ,EAAuB,OAAvB,C;MACL,mC;IACJ,C;;MAGI,oCAAa,2BAAkB,K
AAIB,EAAyB,YAAzB,C;MAEb,OAAO,wBAAK,mBAAY,KAAZ,IAAL,C;IACX,C;;MAGI,oCAAa,2BAAkB,KA
AlB,EAAyB,YAAzB,C;MAEb,aAAa,WAAK,kBAAS,mBAAY,KAAZ,IAAT,C;MACIB,mC;MACA,OAAO,M;IA
CX,C;;MAGI,oCAAa,2BAAkB,KAAIB,EAAyB,YAAzB,C;MAEb,OAAO,WAAK,aAAI,mBAAY,KAAZ,IAAJ,E
AAuB,OAAvB,C;IACbB,C;;MAE+B,mB;IAAA,C;;MC1LgD,sB;MA4BnF,uBAAoC,I;MA2CpC,yBAA6C,I;IAv
EK,C;;MAKZ,wB;MAGIc,gBAAqB,K;IAHY,C;;MAAC,yB;IAAA,C;;MAKJ,oB;IAAA,C;;MAG1B,eAAe,IAAK,
S;MACpB,gBAAc,Q;MACd,OAAO,Q;IACX,C;;MAE+B,iEAAc,IAAd,C;IAAA,C;;MACG,iEAAc,IAAd,C;IAAA,
C;;MACU,+DAAY,IAAZ,EAAkB,KAAIB,C;IAAA,C;;MAd5C,wE;MAAsC,2CAAK,KAAM,IAAX,EAAgB,KA
AM,MAAtB,C;MAAtC,Y;IAAA,C;;MAmBA,YAAQ,Q;IACZ,C;;MAMoB,sD;MAAS,6B;IAAT,C;;MACoC,MAA
M,qCAA8B,8BAA9B,C;IAAN,C;;MAEpC,4BAAwB,Q;IAC5B,C;;MAEsD,sDAAY,OAAZ,C;IAAA,C;;MAI3C,k
D;IAAA,C;;MAC+B,OAAA,0BAAc,U;IAAd,C;;MACT,OAAA,0BAAc,OAAO,I;IAArB,C;;MACD,0BAAc,S;IAA
d,C;;MAJ5B,oBAAoB,oCAAQ,W;MAC5B,6G;IAKJ,C;;MAGI,IAAI,+CAAY,OAAZ,CAAJ,C;QACI,4BAAwB,c
AAO,OAApC;QACxB,OAAO,I;;MAEX,OAAO,K;IACX,C;;MAE+B,OAAA,4BAAwB,K;IAAxB,C;;MA1BvC,I
AAI,4BAAJ,C;QACI,6F;;MA4BJ,OAAO,mC;IACX,C;;MAKqB,Q;MAAA,OAAA,InE2O2D,QAAQ,W;MmE3Ox
F,OAAqB,cAArB,C;QAAqB,wB;QAAf,UnE8JsD,U;QmE9JjD,YnE2KiD,Y;QmE1KxD,iBAAI,GAJJ,EAAS,KAA

T,C;;IAER,C;;MAMsB,sD;MAAS,oC;IAAT,C;;MACkC,MAAM,qCAA8B,gCAA9B,C;IAAN,C;;MACjB,4BAAw
B,Q;IAAxB,C;;MAE+B,wDAAC,OAAd,C;IAAA,C;;MAI3C,kD;IAAA,C;;MAC+B,OAAA,0BAAC,U;IAAd,C;;M
ACT,OAAA,0BAAC,OOAO,M;IAArB,C;;MACD,0BAAC,S;IAAd,C;;MAJ5B,oBAAoB,oCAAQ,W;MAC5B,+G;I
AKJ,C;;MAE+B,OAAA,4BAAwB,K;IAAxB,C;;MAI3B,IAAI,SAAS,KAAb,C;QAAoB,OOAO,I;MAC3B,IAAI,iC
AAJ,C;QAA6B,OOAO,K;MACpC,OOAO,oCAAa,uBAAC,IAAd,EAAoB,KAApB,C;IACxB,C;;MAE+B,OAAA,o
CAAa,yBAAGb,IAAhB,C;IAAb,C;;MAzBvC,IAAI,8BAAJ,C;QACI,iG;;MA2BJ,OOAO,qC;IACX,C;;MAGA,W
AAW,YAAQ,W;MACnB,OOAO,IAAK,UAAZ,C;QACI,YAAY,IAAK,O;QACjB,QAAQ,KAAM,I;QACd,IAAI,Y
AAO,CAAP,CAAJ,C;UACI,YAAY,KAAM,M;UACIB,IAAK,S;UACL,OOAO,K;;MAGf,OOAO,I;IACX,C;;MC
7HgF,oC;IAAjC,C;;MAQ3C,IAAI,UAAU,IAAd,C;QAAoB,OOAO,I;MAC3B,IAAI,0BAAJ,C;QAAsB,OOAO,K;
MAC7B,OOAO,mCAAY,mBAAU,IAAV,EAAGb,KAAhB,C;IACvB,C;;MAK+B,OAAA,mCAAY,2BAAkB,IAAI
B,C;IAAZ,C;;MCb0D,8B;MAAIC,4B;IAArB,C;;IAoBF,C;;IAGoB,C;;MAEd,OAAA,oBAAM,O;IAAN,C;;MAEG
,Q;MAAA,oCAAM,0BAAW,KAAX,CAAN,4D;IAAA,C;;MAI9B,IAAa,I;MAFpB,0BAAW,KAAX,C;MAEoB,gB
AAb,qBAAM,KAAN,C;MAAqB,qC;MAA5B,OOAO,CAAa,O3DsCjB,S2DtCI,2D;IACX,C;;MAGU,oBAAY,MA
AK,OAAL,C;MACIB,qC;MACA,OOAO,I;IACX,C;;MAGU,oBAAY,QAAO,mCAAoB,KAApB,CAAP,EAAMc,C
AAnc,EAAsC,OAAtC,C;MACIB,qC;IACJ,C;;MAGI,IAAI,QAAS,UAAb,C;QAAwB,OOAO,K;MAE/B,uBAAA,o
B9D8vCoB,QM7yC0C,YwD+CrD,QxD/CqD,CN6yC1C,C;M8D7vCpB,qC;MACA,OOAO,I;IACX,C;;MAGI,mCA
AoB,KAApB,C;MAEA,IAAI,UAAS,SAAb,C;QAAmB,OOAO,oBAAO,QAAP,C;MAC1B,IAAI,QAAS,UAAb,C;
QAAwB,OOAO,K;MAE3B,IADE,KACF,e;QAAQ,OOAO,oBAAO,QAAP,C;WACf,IAFE,KAef,O;QAAK,uBxD
3DqD,YwD2D7C,QxD3D6C,CN6yC1C,Q8DlvC6B,oB9DkvC7B,C;;Q8DjvCR,uBAAoC,cAA5B,oBAA4B,EA
AV,CAAU,EAAP,KAAO,CAAY,QxD5DE,YwD4DK,QxD5DL,CwD4DF,EA44C,cAAN,oBAAM,EAAY,KAAZ,EA
AmB,SAAnB,CAA5C,C;;MAG5D,qC;MACA,OOAO,I;IACX,C;;MAGI,0BAAW,KAAX,C;MACA,qC;MACA,O
AAW,UAAS,sBAAb,GACG,oBAAY,MADf,GAGG,oBAAY,QAAO,KAAP,EAAC,CAAd,CAAIB,CAAMc,CAAn
C,C;IACR,C;;MAGkB,Q;MAAA,2B;MAAd,mD;QACI,IAAI,4BAAM,KAAN,GAAgB,OAAb,CAAJ,C;UACU,o
BAAY,QAAO,KAAP,EAAC,CAAd,C;UACIB,qC;UACA,OOAO,I;;MAGf,OOAO,K;IACX,C;;MAGI,qC;MACM,
oBAAY,QAAO,SAAP,EAAb,UAAU,SAAV,IAAIB,C;IACtB,C;;MAGI,uBpBhGuC,E;MoBiGvC,qC;IACJ,C;;M
AG+C,OAAM,QAAN,oBAAM,EAQ,OAAR,C;IAAN,C;;MAEI,OAAM,YAAN,oBAAM,EAAY,OAAs,C;IAAN
,C;;MAEzB,qBAAC,oBAAd,C;IAAA,C;;MACY,OAAA,EAAS,MAAM,MAAK,oBAAL,C;IAAf,C;;MAIC,oCAA
a,kCAAYB,SAAZB,C;MADoB,Y;IAAA,C;;MAKjC,oCAAa,mCAA0B,SA1B,C;MAD6B,Y;IAAA,C;;MA7G9C,
mD;MAG8B,sBpBPa,EoBOb,C;MAH9B,Y;IAAA,C;;MAU0B,+B;QAAA,kBAAuB,C;MALjD,mD;MAKsD,sBpB
dX,EoBcW,C;MALtD,Y;IAAA,C;;MAOA,mD;MAGqD,sBxDhBa,YwDgBR,QxDhBQ,CwDgBb,C;MAHrD,Y;IA
AA,C;;MCfgB,UACM,M;MAFtB,aAAa,C;MACb,wBAAGb,SAAhB,gB;QAAgB,cAAA,SAAhB,M;QAEQ,oB;UA
AmB,U;;UACnB,IhBHiC,MAAa,YgBGnC,OhBHmC,CgBG9C,C;YAAwD,iCAAhC,OAAGC,C;;gBAExD,uC;YA
AmC,2BAAR,OAAQ,C;eACnC,wC;YAAmC,2BAAR,OAAQ,C;eACnC,sC;YAAmC,2BAAR,OAAQ,C;eACnC,u
C;YAAmC,2BAAR,OAAQ,C;;YAEA,kBAAR,OAAQ,C;;QATvC,wB;QAYA,SAAS,MAAK,MAAL,QAAc,WAA
d,I;;MAEb,OOAO,M;IACX,C;;M7DhBsG,4C;IAAtG,C;IAAA,6D;MAC4C,+BAAW,CAAX,EAAC,CAAd,C;IAA
A,C;IAD5C,kF;;M8DLI,IAAI,6BAAJ,C;QACU,KAAY,MAAK,UAAAL,C;;QAEIB,UAAU,KAAV,EAawC,CAAx
C,EAaiD,cAAN,KAAM,CAAjD,+BAAuE,UAAvE,E;;IAER,C;;MAIyB,uB;QAAgB,OAAA,kBAAW,SAAQ,CAA
R,EAAW,CAAX,C;MAA3B,C;IAAA,C;;MADrB,IAAI,6BAAJ,C;QACI,iBAaiB,gC;QACX,KAAY,MAAK,UAA
L,C;;QAEIB,UAAU,KAAV,EAawC,CAAxC,EAaiD,cAAN,KAAM,CAAjD,EA44D,UAA5D,C;;IAER,C;;MAIy
C,OAAE,iBAAF,CAAE,EAU,CAAV,C;IAAa,C;;MADpD,IAAI,6BAAJ,C;QACI,iBAaiB,gB;QACX,KAAY,MA
AK,UAAAL,C;;QAEIB,UAAU,KAAV,EAawC,CAAxC,EAaiD,cAAN,KAAM,CAAjD,EA44D,cAA5D,C;;IAER,
C;;MAWyC,QAAC,IAAM,CAAP,KAAa,IAAM,CAAnB,K;IAAsB,C;;MAP3D,sC;QAAiC,OAAjC,yB;;MACA,4
BAA4B,K;MAE5B,YAAY,E;MAGZ,iBAAC,CAAd,UAAsB,GAAtB,U;QAAiC,KAAY,MAAK,KAAL,C;MAC7C,
iBAaiB,kC;MACX,KAAY,MAAK,UAAAL,C;MACIB,mBAAC,CAAd,YAAsB,KAAM,OAAsB,Y;QACI,QAAQ,M
AAM,UAAQ,CAAR,IAAN,C;QACR,QAAQ,MAAM,OAAN,C;QACR,IAAI,CAAC,IAAM,CAAP,OAAC,IAAM,
CAApB,KAA0B,KAAK,CAAnC,C;UAAsC,OOAO,K;;MAEjD,4BAA4B,I;MAC5B,OOAO,I;IACX,C;;MAII,aAA
a,gBAAMB,KAAM,OAAsB,O;MACb,aAAa,YAAU,KAAV,EAaiB,MAAjB,EAAYB,KAAsB,EAAGC,YAAhC,E
AA8C,UAA9C,C;MACb,IAAI,WAAW,KAAf,C;Q9EspVa,gB;QADb,YAAY,C;QACZ,wB8ErpVI,M9EqpVJ,gB;U

AAa,W8ErpVT,M9EqpVJ,M;U8ErpVoC,M9EqpVV,c8ErpVU,E9EqpVV,sB8ErpVU,Y9EqpVD,I;;;I8EnpVvC,C;;
MAII,IAAI,UAAS,GAAb,C;QACI,OAAO,K;;MAGX,aAAa,CAAC,QAAQ,GAAR,IAAD,IAAgB,CAAhB,I;MAC
b,WAAW,YAAU,KAAV,EAAiB,MAAjB,EAAyB,KAAzB,EAAgC,MAAhC,EAAwC,UAAxC,C;MACX,YAAY,
YAAU,KAAV,EAAiB,MAAjB,EAAyB,SAAS,CAAT,IAAzB,EAAqC,GAAR,C,EAA0C,UAA1C,C;MAEZ,aAAiB,
SAAS,MAAb,GAAqB,KAArB,GAAgC,M;MAG7C,gBAAgB,K;MACHB,iBAAiB,SAAS,CAAT,I;MACjB,aAAU,
KAAV,OAAiB,GAAjB,M;QAEQ,iBAAa,MAAb,IAAuB,cAAc,GAAR,C,C;UACI,gBAAgB,KAAK,SAAL,C;UACH
B,iBAAiB,MAAM,UAAN,C;UAEjB,IAAI,UAAW,SAAQ,SAAR,EAAMB,UAAAnB,CAAX,IAA6C,CAAjD,C;YA
CI,OAAO,CAAP,IAAY,S;YACZ,6B;;;YAEA,OAAO,CAAP,IAAY,U;YACZ,+B;;;cAGR,iBAAa,MAAb,C;UACI,
OAAO,CAAP,IAAY,KAAK,SAAL,C;UACZ,6B;;;UGA,OAAO,CAAP,IAAY,MAAM,UAAN,C;UACZ,+B;;;M
AMZ,OAAO,M;IACX,C;;;MCpFI,2C;IAAA,C;;MAC+D,sBAAU,MAAV,C;IAAA,C;;MAEd,Q;MAAA,wEAAqB
,C;IAArB,C;;;MAHjD,kD;QAAA,iC;;MAAA,2C;IAAA,C;;MCuBA,iC;MAKA,8B;MA8CA,0BAAMe,I;;;MA1En
E,oB;MAA+B,6B;IAA/B,C;;MAE6D,MAAM,qCAA8B,iCAA9B,C;IAAN,C;;MAErD,WAAa,Q;IACjB,C;;MAEuE
,yCAAc,OAAc,C;IAAA,C;;MAEC,OAAA,iCAAY,W;IAAZ,C;;MAGpE,IAAI,oBAAS,OAAT,CAAJ,C;QACI,WA
Aa,cAAO,OAAQ,IAAf,C;QACb,OAAO,I;;MAEX,OAAO,K;IACX,C;;MAE+B,OAAA,WAAa,K;IAAb,C;;MA+C
/B,0BAAY,Q;IAEHb,C;;MAEmD,OAAA,0BAAY,gBAAS,GAAT,C;IAAZ,C;;MAEgB,gBAAZ,0B;MAAY,c;;Q7
D23CnD,Q;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,aAAO,K;UAAP,e;;QACrB,2B;QAAbB,OAAGB,cAAhB
,C;UAAgB,yB;UAAM,I6D33CmD,uBAAS,gB7D23C9C,O6D33CwD,MAAV,Q7D23C5D,C;YAAwB,aAAO,I;YA
AP,e;;QAC9C,aAAO,K;;;M6D53CgD,iB;IAAA,C;;MAK/C,IAAI,+BAAJ,C;QACI,0BAAW,qB;;MAEf,OAAO,sC
;IACX,C;;MAE6E,iC;IAAA,C;;MAEIC,OAAA,0BAAY,WAAI,GAAJ,C;IAAZ,C;;MAEC,OAAA,0BAAY,aAAI,G
AAJ,EAAS,KAAT,C;IAAZ,C;;MAEP,OAAA,0BAAY,cAAO,GAAP,C;IAAZ,C;;MAEH,OAAA,0BAAY,K;IAAZ,
C;;MA7DtC,iD;MAAuD,8B;MArC3D,mB;MAScQ,8BAAMb,W;MACnB,2BAAgB,WAAy,S;MAFhC,Y;IAAA,
C;;MAKA,iD;MAGuB,aAAK,kEAAAL,Q;MAHvB,Y;IAAA,C;;MAcyC,0B;QAAA,aAAoB,G;MAT7D,iD;MASqE,
qB;MnElCrE,IAAI,EmEoCQ,mBAAMb,CnEpC3B,CAAJ,C;QACI,cmEmCgC,+C;QnElChC,MAAM,gCAAYB,OA
AQ,WAAjC,C;;MAFV,IAAI,EmEqCQ,cAAc,CnErCtB,CAAJ,C;QACI,gBmEoC2B,yC;QnEnC3B,MAAM,gCAAY
B,SAAQ,WAAjC,C;;MmEuBV,Y;IAAA,C;;MAeA,iD;MAA2C,eAAK,eAAL,EAASB,GAATB,Q;MAA3C,Y;IAAA
,C;;MAGA,iD;MAG8C,qB;MAC1C,KAAK,gBAAO,QAAP,C;MAJT,Y;IAAA,C;;MA0C0E,gBAAnE,aAAmB,gE
AAAnB,C;MAA2E,wB;MAAIF,O/DrCO,S;I+DsCX,C;;MCzGI,yB;;;MA2CI,UAAU,kBAAI,aAAI,OAAJ,EAAa,IAA
b,C;MACd,OAAO,W;IACX,C;;MAGI,kBAAI,Q;IACR,C;;MAM6D,OAAA,kBAAI,mBAAY,OAAZ,C;IAAJ,C;;M
AEpB,OAAA,kBAAI,U;IAAJ,C;;MAEY,OAAA,kBAAI,KAAK,W;IAAT,C;;MAEH,OAAA,kBAAI,cAAO,OAAP
,CAAJ,Q;IAAA,C;;MAEZ,OAAA,kBAAI,K;IAAJ,C;;MA7DtC,iD;MAGoB,8B;MAVxB,mB;MAWQ,sBAAM,gB
;MAJV,Y;IAAA,C;;MAOA,iD;MAG2C,8B;MAjB/C,mB;MAkBQ,sBAAM,eAAgB,QAAS,KAAzB,C;MACN,qBA
AO,QAAP,C;MALJ,Y;IAAA,C;;MAiByC,0B;QAAA,aAAoB,G;MAT7D,iD;MASkE,8B;MA/BtE,mB;MAGcQ,sB
AAM,eAAgB,eAAhB,EAAiC,UAAjC,C;MAVV,Y;IAAA,C;;MAaA,iD;MAA2C,eAAK,eAAL,EAASB,GAATB,Q;
MAA3C,Y;IAAA,C;;MAEA,iD;MAM0C,8B;MA3C9C,mB;MA4CQ,sBAAW,G;MAPf,Y;IAAA,C;;MAwCmC,gB
AA5B,eAAQ,eAAR,C;MAAoC,6B;MAA3C,OhELO,S;IlgEMX,C;;MC/DyC,kC;MAErC,oBAAKC,kB;MACIC,sB
AAyB,C;IAHW,C;;MAAC,8B;IAAA,C;;MAGrC,0B;IAAA,C;MAAA,0B;IAAA,C;;MAII,eAAe,aAAS,qBAAY,G
AAZ,C;MACxB,mBAAMb,6BAASB,QAATB,C;MACnB,IAAI,oBAAJ,C;QAEI,kBAAW,QAAX,IAAuB,mCAAY,
GAAZ,EAAiB,KAAjB,C;;;QAEvB,IAAI,6BAAJ,C;UAEI,YAA+B,Y;UAC/B,IAAI,aAAS,gBAAO,KAAM,IAAb,
EAAkB,GAAIB,CAAb,C;YACI,OAAO,KAAM,gBAAS,KAAT,C;;;YAEb,kBAAW,QAAX,IAAuB,CAAQ,KAAR
,EAAe,mCAAY,GAAZ,EAAiB,KAAjB,CAAf,C;YACvB,6B;YACA,OAAO,I;;;UAIX,YAAuC,Y;UACvC,cAAkB
,wBAAN,KAAM,EAAiB,GAAjB,C;UACIB,IAAI,eAAJ,C;YACI,OAAO,OAAM,gBAAS,KAAT,C;;UAEX,KAAy
,MAAK,mCAAY,GAAZ,EAAiB,KAAjB,CAAL,C;;;MAG1B,6B;MAEA,OAAO,I;IACX,C;;MAIuB,Q;MADnB,e
AAe,aAAS,qBAAY,GAAZ,C;MACL,oCAAsB,QAATB,C;MAAA,iB;QAAMC,OAAO,I;;MAA7D,mBAAMb,I;MA
CnB,IAAI,6BAAJ,C;QACI,YAAgC,Y;QAChC,IAAI,aAAS,gBAAO,KAAM,IAAb,EAAkB,GAAIB,CAAb,C;UpB
zDR,OoB0D6B,iBpB1DvB,CoB0DmC,QpB1DnC,C;UoB2DM,6B;UACA,OAAO,KAAM,M;;;UAEb,OAAO,I;;;Q
AGX,YAAuC,Y;QACvC,8BAAc,KAAAd,iB;UACI,cAAY,MAAM,KAAN,C;UACZ,IAAI,aAAS,gBAAO,GAAP,E
AAY,OAAM,IAAIB,CAAb,C;YACI,IAAI,KAAM,OAAN,KAAc,CAAIB,C;cACU,KAAN,UAA2B,C;cpBtE/C,Oo
BwEqC,iBpBxE/B,CoBwE2C,QpBxE3C,C;;;coB2EoB,KAAy,QAAO,KAAP,EAAC,CAAd,C;;YAEtB,6B;YAEA,

OAAO,OAAM,M;;;MAIzB,OAAO,I;IACX,C;;MAGI,oBAAa,kB;MACb,YAAO,C;IACX,C;;MAEyC,uBAAS,GAAT,S;IAAA,C;;MAEV,Q;MAAA,+BAAS,GAAT,8B;IAAA,C;;MAGR,Q;MAAA,oCAAsB,aAAS,qBAAY,GAAZ,CAA/B,C;MAAA,iB;QAAoD,OAAO,I;;MAA9E,mBAAmB,I;MACnB,IAAI,6BAAJ,C;QACI,YAAgC,Y;QAChC,IAAI,aAAS,gBAAO,KAAM,IAAb,EAaKb,GAAIB,CAAb,C;UACI,OAAO,K;;;UAEP,OAAO,I;;;QAGX,YAAuC,Y;QACvC,OAAa,wBAAN,KAAM,EAAiB,GAAjB,C;;IAErB,C;;MAGI,sB;;QIFg7BY,Q;QAaHb,iD;UAAgB,cAAhB,e;UAAsB,lkFh7BK,aAAS,gBIFg7BA,OkFh7Ba,IAAb,MIFg7Bd,C;YAAwB,qBAAO,O;YAAP,uB;;;QAC9C,qBAAO,I;;;MkFj7BH,yB;IAAA,C;;MAIO,wD;MACH,aAAY,E;MAEZ,YAA0B,MAAa,MAAK,qCAAL,C;MACvC,gBA Ae,E;MAEf,oBAA4B,I;MAC5B,eAAc,K;MACd,iBAAgB,E;MACHb,iBAAqC,I;IATIC,C;;MAYC,IAAI,6BAAwB,YAA5B,C;QACI,gBAAqB,iBAAqD,O;QAC1E,IAAI,4DAAc,SAaIB,C;UACI,OAAO,C;;MAGf,IAAI,yDAAa,SAAK,OAAiB,C;QACI,oBAAe,2CAAW,UAAK,aAAL,CAAX,C;QACf,eAAU,iC;QACV,iBAAY,C;QACZ,OAAO,C;;;QAEP,oBAAe,I;QACf,OAAO,C;;IAEf,C;;MAGI,IAAI,eAAS,EAAb,C;QACI,aAAQ,oB;MACZ,OAAO,eAAS,C;IACpB,C;;MAIoB,Q;MADhB,IAAI,CAAC,cAAL,C;QAAGB,MAAM,6B;MACN,IAAI,YAAJ,C;QACZ,yBAAqD,cAArD,C;;;QAEa,OAAb,iB;;MAHJ,oB;MAKA,iBAAiB,S;MACjB,aAAQ,E;MACR,OAAO,S;IACX,C;;MrE7CR,IqEgDyB,crEhDrB,QAAJ,C;QACI,cAhByB,0B;QAIbZB,MAAM,6BAAsB,OAAQ,WAA9B,C;;MqE+CE,6BAAYB,cAAO,6BAAY,IAAnB,C;MACzB,iBAAy,I;MAEZ,uC;IACJ,C;;;MARDJ,2D;IAUDJ,C;;MAGI,mBAAmB,kBAAW,QAAX,C;MACnB,OAAW,iBAAiB,SAArB,GAAgC,IAAhC,GAA0C,Y;IACrD,C;;;MCTKI,aAAR,MAAM,OAAe,CAAP,IAAO,C;MAEb,OAAO,KAAP,IAAgB,C;MrBXpB,OqBYqB,MrBZf,CqBYuB,KrBzvB,C;MqBaF,OA AO,M;IACX,C;;;MCPmC,kC;MAEnC,oBAAkC,kB;MACIC,sBAAyB,C;IAHS,C;;MAAC,8B;IAAA,C;;MAGnC,0B;IAAA,C;MAAA,0B;IAAA,C;;MAYI,IAAI,0BAAJ,C;QAAoB,OAAO,K;MAC3B,OAAO,kBAAW,GAAX,MAAoB,S;IAC/B,C;;MAGI,IAAI,0BAAJ,C;QAAoB,OAAO,I;MAC3B,YAAy,kBAAW,GAAX,C;MACZ,OAAW,UAAU,SAArB,GAAgC,KAAhC,GAA2D,I;IAC/D,C;;MvEPA,IAAI,EUWQ,uBvEXR,CAAJ,C;QACI,cAda,qB;QAeb,MAAM,gCAAYB,OAAQ,WAAjC,C;;MuEUN,eAAe,kBAAW,GAAX,C;MACf,kBAAW,GAAX,IAAKB,K;MAEIB,IAAI,aAAa,SAAjB,C;QACI,6B;QAEA,OAAO,I;;;QAGP,OAAO,Q;;IAEf,C;;MAGI,IAAI,0BAAJ,C;QAAoB,OA AO,I;MAC3B,YAAy,kBAAW,GAAX,C;MACZ,IAAI,UAAU,SAAd,C;QtBnDJ,OsBoDyB,iBtBpDnB,CsBoD+B,GtBpD/B,C;QsBqDE,6B;QAEA,OAAO,K;;;QAGP,OAAO,I;;IAEf,C;;MAII,oBAAa,kB;MACb,YAAO,C;IACX,C;;MAIW,oD;MACH,cAAkC,MAAa,MAAK,mCAAL,C;MAC/C,kBAA4B,qBAAL,WAAK,C;MAC5B,iBAA+B,I;IAH5B,C;;MAK+B,OAAA,eAAS,U;IAAT,C;;MAMX,gB;MAHnB,UAAU,eAAS,O;MACnB,iBAAU,G;MAES,+E;MAAnB,OAAO,iD;IACX,C;;MAIkC,UAA9B,M;MAAA,oC;MAA8B,YAAa,c;MvEchD,uB;MAeP,IAfoB,KAehB,QAAJ,C;QACI,cAhByB,0B;QAIbZB,MAAM,6BAAsB,OAAQ,WAA9B,C;;;QAEN,sBAnBgB,K;;MuEde,oBAAO,sFAAP,C;IAC3B,C;;;MAjBJ,yD;IAMBJ,C;;MAEsD,8B;MAAA,oD;IAAA,C;;MACTb,uB;IAAA,C;;MACE,6CAAUB,gBAAvB,C;IAAA,C;;MAEU,OAAA,2BAAUb,aAAI,gBAAJ,EAAS,QAAT,C;IAAvB,C;;MAET,OAAA,mCAAY,uBAAc,IAAd,C;IAAZ,C;;MACG,OAAA,mCAAY,uBAAc,IAAd,C;IAAZ,C;;MACU,OAAA,mCAAY,qBAAY,IAAZ,EAaKb,KAAIB,C;IAAZ,C;;;MARM,iE;IAAA,C;;;MCNtD,sBAOsC,I;MA6CtC,yB;;;MARHmD,0CAAqC,GAArC,EAA0C,KAA1C,C;MAC/C,oBAAuC,I;MACvC,oBAAuC,I;IAFb,C;;;MAK9B,oB;MAA+B,6B;IAA/B,C;;MAEI,oB;MACI,cACsC,I;MAEtC,cACsC,I;MAGIC,cAAO,iC;IARf,C;;MAaQ,OAAO,gBAAS,I;IACpB,C;;MAII,IAAI,CAAC,cAAL,C;QAAGB,MAAM,6B;MAEtB,cAAc,0B;MACd,cAAO,O;MACa,gBAAb,OAAQ,a;MAAf,cpE8DS,SoE9DoB,KAAO,iCpE8DzC,GAAqB,SAArB,GAA+B,I;MoE7D1B,OAAO,O;IACX,C;;MxEgCR,IAAI,EwE7Bc,eAAQ,IxE6tB,CAAJ,C;QACI,cAdW,e;QAeX,MAAM,6BAAsB,OAAQ,WAA9B,C;;MwE5BS,oCAAP,0BAAO,C;MACP,gCAAI,cAAO,0BAAO,IAAd,C;MAEJ,cAAO,I;IACX,C;;;MAGqD,MAAM,qCAA8B,iCAA9B,C;IAAN,C;;MAErD,WAAmB,Q;IACvB,C;;MAEuE,yCAAc,OAAd,C;IAAA,C;;MAEC,qD;IAAA,C;;MAGpE,IAAI,oBAS,OAAT,CAAJ,C;QACI,WAAmB,cAAO,OAAQ,IAAf,C;QACnB,OAAO,I;;MAEX,OAAO,K;IACX,C;;MAE+B,OAAA,WAAmB,K;IAAnB,C;;;MxEEEnC,IAAI,EwEgBM,0BAAQ,IAAR,IAAgB,0BAAQ,IxEhB9B,CAAJ,C;QACI,cAdW,e;QAeX,MAAM,6BAAsB,OAAQ,WAA9B,C;;MwEgBN,YAAy,mB;MACZ,IAAI,SAAS,IAAb,C;QACI,sBAAO,S;QACP,yBAAO,S;QACP,yBAAO,S;;;QAGK,YAAa,KAAM,a;QxERhC,uB;QAeP,IAfoB,KAehB,QAAJ,C;UACI,gBAhByB,0B;UAIbZB,MAAM,6BAAsB,SAAQ,WAA9B,C;;;UAEN,sBAnBgB,K;;QwEQZ,+B;QAEA,yBAAO,K;QACP,yBAAO,K;QAEP,qBAAa,S;QACb,qBAAa,S;;IAErB,C;;MAMI,IAAI,SAAK,aAAL,KAAC,SAaIB,C;QAEI,sBAAO,I;;;QAEP,IAAI,wBAAS,SAAb,C;UAEI,sBAAO,sB;;QAEX,qDAAc,sB;QACd,qDAAc,sB;;MAEIB,yBAAO,I;MACP,yBAAO,I;IACX,C;;MA6CI,kBAAL,Q;MACJ,sBAAO,I;IACX,C;;MAQmD,OAAA,kBAAL,

mBAAY,GAAZ,C;IAAJ,C;;MAGIB,Q;MAAA,0B;MAAA,iB;QAAQ,OAAO,K;;MAA5C,WAA6B,I;;QAEzB,IAA
I,OAAA,IAAK,MAAL,EAAC,KAAc,CAAJ,C;UACI,OAAO,I;;QAEEX,OAAO,cAAA,IAAK,aAAL,C;;MACF,iBA
AS,mBAAT,C;MACT,OAAO,K;IACX,C;;MAG2E,uC;IAAA,C;;MAEXB,Q;MAAJ,QAAL,OAAJ,kBAAL,WAAI,G
AAJ,CAAJ,6B;IAAA,C;;MAG3C,UAAU,kBAAL,WAAI,GAJJ,C;MACd,IAAI,OAAO,IAAX,C;QACI,eAAe,6BA
AW,GAAX,EAAGB,KAAhB,C;QACf,kBAAL,aAAI,GAJJ,EAAS,QAAT,C;QACK,wBAAT,QAAS,C;QACT,OA
AO,I;;QAEp,OAAO,GAAL,gBAAS,KAAT,C;;IAEnB,C;;MAGI,YAAAY,kBAAL,cAAO,GAAP,C;MACHB,IAAI,S
AAS,IAAb,C;QACU,sBAAN,KAAM,C;QACN,OAAO,KAAM,M;;MAEjB,OAAO,I;IACX,C;;MAESc,OAAA,kB
AAL,K;IAAJ,C;;MArFtC,uD;MAGuB,qB;MAnJ3B,yB;MAoJQ,sBAAM,gB;MAJv,Y;IAAA,C;;MAOA,uD;MAA
oD,qB;MAvJxD,yB;MAyJc,Q;MAAN,sBAAM,+D;MAFV,Y;IAAA,C;;MAcyC,0B;QAAA,aAAoB,G;MAT7D,uD;
MASqE,eAAM,eAAN,EAAbB,UAAvB,Q;MArKzE,yB;MASkQ,sBAAM,gB;MAVV,Y;IAAA,C;;MAaA,uD;MA
A2C,qBAAK,eAAL,EAAsB,GAAtB,Q;MAA3C,Y;IAAA,C;;MAEA,uD;MAG2C,qB;MA9K/C,yB;MA+KQ,sBA
AM,gB;MACN,KAAK,gBAAO,QAAP,C;MALT,Y;IAAA,C;;MAmEoD,gBAA7C,qBAAYB,eAAzB,C;MAAqD,wB;
MAA5D,OpEvKO,S;IoEwKX,C;;;MCzOI,uD;MAAmD,eAAM,GAAN,Q;MAPvD,yB;MAOI,Y;IAAA,C;;MAEA
,uD;MAGuB,eAAM,oBAAN,Q;MAZ3B,yB;MASI,Y;IAAA,C;;MAKA,uD;MAG8C,eAAM,oBAAN,Q;MAjBID,y
B;MAkBQ,qBAAO,QAAP,C;MAJJ,Y;IAAA,C;;MAGByC,0B;QAAA,aAAoB,G;MAT7D,uD;MASqE,eAAM,qBA
AsB,eAAtB,EAAC,UAAvC,CAAN,Q;MA9BzE,yB;MAqBI,Y;IAAA,C;;MAWA,uD;MAA2C,qBAAK,eAAL,EA
AsB,GAAtB,Q;MAA3C,Y;IAAA,C;;MAa+C,gBAAXC,mBAAC,qBAAd,C;MAAGD,6B;MAAvD,OrE2BO,S;IqE1
BX,C;;;gFtE5CA,yB;MAAA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EA
Ac,CAAd,C;MAAA,C;MAD5C,kF;MAAA,6B;QAAsG,+C;MAAA,C;KAAtG,C;;MuEQI,OAAO,O;IACX,C;;ICfA,
C;;MAEQ,mBAAM,IAAN,C;IACJ,C;;MAGI,mBAAM,OAAN,C;MACA,c;IACJ,C;;IAIkB,C;;MAKMc,qB;MAA
7B,gC;IAAD,C;;MAGnB,oBA+DyC,OA/Dd,OA+Dc,C;MA9DzC,iBAAa,OAAM,aAAN,C;IACjB,C;;MAKgC,qB;
IAFpC,C;;MAIQ,OAAQ,KAAL,OAAJ,C;IACZ,C;;MAGI,OAAQ,KAAL,OAAJ,C;IACZ,C;;MAGI,OAAQ,KAAL,E
AAJ,C;IACZ,C;;MAKiC,qB;MACjC,cAAa,E;IAHjB,C;;MAMQ,eAoCyC,OApCxB,OAoCwB,C;IANc7C,C;;MA
GI,cAAS,E;IACb,C;;MAKwC,yB;IAF5C,C;;MAIQ,QAwByC,OAxBI,OAwb0B,C;MAvBzC,QAAU,eAAF,CA
AE,EAAY,EAZC,C;MACV,IAAI,KAAK,CAAT,C;QACI,4BAAU,C1DiI0E,W0DjI9D,C1DiI8D,E0DjI3D,C1DiI2
D,C;Q0DhIpF,Y;QACA,IAAI,C1D4HiE,W0D5HrD,IAAI,CAAJ,I1D4HqD,C;;M0D1HzE,4BAAU,C;IACd,C;;MA
GI,OAAQ,KAAL,WAAJ,C;MACR,cAAS,E;IACb,C;;;MAW6C,cAAa,KAAb,C;IAAA,C;;MAI7C,MAAO,U;IACX
,C;;MAII,MAAO,iBAAQ,OAAR,C;IACX,C;;MAII,MAAO,eAAM,OAAN,C;IACX,C;;M7BzCI,sC;MAAA,4C;IA
TJ,C;IAAA,yF;MAWob,2B;IAAA,CAXpB,E;IAAA,4E;MAcY,wBAAW,MAAX,C;IAAA,C;IAdZ,sF;;MDiHW,Q;
MADP,e+B1JA,M/B0JA,C;MACO,Q+B3JP,M/B2JO,+D;M+B1JX,W;IAAA,C;;MCAI,0B;MASA,gBAA2B,a;IA
V/B,C;;MAQgB,OAAA,eAAS,Q;IAAT,C;;MAKR,UAAU,IAAK,S;MAEX,YAAQ,2CAAR,C;QACI,gBAAC,MAA
O,M;WAEzB,YAAQ,yBAAR,C;QACI,gBAAC,yC;QACd,eAAS,oBAAW,MAAX,C;;QAEI,MAAM,6BAAsB,iB
AAtB,C;IAEtB,C;;MASW,Q;MALP,IAAI,kBAAW,2CAAf,C;QACI,gBAAS,yB;QACT,OAAO,yB;;MAEX,aAAa,
IAAK,S;MAEd,eAAW,yCAAX,C;QAAsB,gC;WACtB,0C;QAA4B,MAAM,MAAO,U;;QACjC,a;MAHZ,W;IAKJ,
C;;MAICA,0D;MACyD,6BAAK,QAAL,EAAC,2CAAf,C;MADzD,Y;IAAA,C;;MCqHA,aAAa,IAAb,MAAa,E;M
ACb,KAAK,MAAL,C;MACA,OAAO,M;IACX,C;;MC3HyG,kBAAPB,wBAAC,IAAd,C;MAA0B,I3EoE3G,a;M2E
pEiF,O3EqE1E,W;I2ErE0E,C;;MAQvC,kBAA1B,cAAhB,sCAAAGB,EAAC,IAAd,EAAoB,IAApB,C;MAAiC,sB3E2
E3C,W2E3E2C,C;MAAjD,O3E4EO,W;I2E5EP,C;;MCZ8C,8B;MAAA,OCIN,aDjwB,yBAAa,QAAb,mCCIxB,CjE
6rBgC,sB;IgejsB1B,C;;M7Fu8GvC,kBAAY,gB;MA8EH,Q;MAAhB,wB6F7gHqB,U7F6gHrB,gB;QAAgB,c6F7g
HK,U7F6gHrB,M;QAAsB,IAAI,C6F7gHkB,oB7F6gHP,O6F7gHO,C7F6gHtB,C;UAAyB,WAAy,WAAI,OAAJ,C;
;M6F7gH3D,qB7F8gHO,W;M6F7gHP,IvF4JwD,CuF5JpD,cvF4JqD,UuF5JzD,C;QhEoGuC,U;QgEnGnC,qBhEmG
yD,OAAtB,+BgEnGd,mBhEmGc,uBAAsB,CAAO,W;QkEvGpD,kBAAhB,sB;QFMQ,0C;QACA,IAAI,EhEkNoC,
0BAAU,CgEIN9C,CAAJ,C;UACI,2BAAO,GAAP,C;;QAEW,sCAAA,GAAb,C;QALnB,sB5E8DG,W8EnE8B,W;Q
FYjC,OAAO,I;;MAGX,OAAO,K;IACX,C;;MAQmB,c;;Q7Fk7TC,Q;QAAhB,wB6F7TI,U7Fk7TJ,gB;UAAgB,c6F
I7TZ,U7Fk7TJ,M;UAAAsB,I6F7TD,oB7Fk7Te,O6F7Tf,C7Fk7TC,C;YAAwB,aAAO,I;YAAP,e;;;QAC9C,aAAO,K
;;M6Fn7TP,e;QACI,kBAA6B,MAAX,UAAW,C;QhE8EM,U;QgE7Eb,ahE6EmC,OAAtB,+BgE7EvB,mBhE6EuB,
uBAAsB,CAAO,W;QgE7EX,kBC5BjB,aD4BD,MC5BC,CjE0uC6C,uBAAzB,CAAYB,C;QVxIb9E,kBAAS,gB;Q
AyEA,U;QAAA,+B;QAAhB,OAAGB,gBAAhB,C;UAAgB,6B;UAAM,I0E/rB4C,4B1E+rB9B,S0E/rB8B,C1E+rB5

C,C;YAAwB,WAAY,WAAL,SAAJ,C;;Q0E/rBtD,sBAAmF,e1EgsBhF,W0EhsBgF,EAAa,GAAb,C;QACnF,OAAO ,I;;MAGX,OAAO,K;IACX,C;;MGICY,8BAAY,IAAK,UAAjB,IAA8B,uBAAY,IAAK,mB;IAA/C,C;;MAMA,8BA AY,IAAK,a;IAAjB,C;;MCZwD,gCAAqB,OAArB,C;IAAA,C;;MAEjC,wB;IAAD,C;;MAE1B,eAAQ,KAAR,C;IA CJ,C;;MAEyC,iCAAuB,cAAvB,M;IAAA,C;;MCCO,8B;MAAS,uB;IAAT,C;;MACjB,OAAA,gBAAY,O;IAAZ,C; ;MAG3B,IADoC,KACpC,IAAG,CAAH,IADoC,KACpC,IAAM,sBAAN,C;QAD8B,OACX,gBAAY,MAAK,KAA L,C;;;QACvB,MAAM,8BAA0B,WAAQ,KAAR,6BAAmC,sBAAnC,MAA1B,C;IAFgB,C;;;MAHc,4C;IAAA,C;;M CJhD,OAAO,yBAAP,C;QACI,sBAAY,mCAAZ,C;;IAER,C;;MAQI,sBAAY,sCAAqB,gBAAe,IAAf,CAA5B,C;M ACA,OAAO,S;IACX,C;;MCb8C,gB;IAAA,C;;MAMwB,OAAK,S;IAAL,C;;MAQG,gB;IAAA,C;;MAmB5D,UAP F,M;MAFP,QAAc,S;MAGV,cAAK,UAAL,U;QACI,mBAAK,UAAL,G;;QACJ,IvC3BqC,MAAa,YuC2BvC,CvC3 BuC,CuC2BID,C;UAC6B,8BAAzB,CAAyB,C;;;UAGN,UAA1B,uDAakB,Y;;MAP3B,a;IASJ,C;;;MCxB4G,Q;MA AA,+BAAS,I;sCAAIB,O,2DAAA,O;;;IAApE,C;;;MAC3B,iD;MAAuB,oBAAK,IAAL,EAAW,IAAX,C;MAC H,0BAAY,MAAK,KAAL,EAAW,IAAX,EAAiB,IAAjB,C;MADhC,Y;IAAA,C;;MAIA,iD;MAAuC,oBAAK,OAA L,EAAc,IAAd,C;MACnB,0BAAY,MAAK,KAAL,EAAW,OAAX,EAAoB,IAApB,C;MADhC,Y;IAAA,C;;MAIA,i D;MAAwC,oBAAK,SAAL,EAAgB,KAahB,C;MACpB,0BAAY,MAAK,KAAL,EAAW,SAAX,EAAsB,KAAtB,C ;MADhC,Y;IAAA,C;;MAM4G,Q;MAAA,+BAAS,I;sCAAIB,O,2DAAA,O;;;IAApE,C;;;MAC/B,mD;MAAu B,sBAAK,IAAL,EAAW,IAAX,C;MACC,4BAAY,MAAK,KAAL,EAAW,IAAX,EAAiB,IAAjB,C;MADpC,Y;IAA A,C;;MAIA,mD;MAAuC,sBAAK,OAAAL,EAAc,IAAd,C;MACf,4BAAY,MAAK,KAAL,EAAW,OAAX,EAAoB,I AApB,C;MADpC,Y;IAAA,C;;MAIA,mD;MAAwC,sBAAK,SAAL,EAAgB,KAahB,C;MACHb,4BAAY,MAAK, KAAL,EAAW,SAAX,EAAsB,KAAtB,C;MADpC,Y;IAAA,C;;MAKgG,qBAAU,OAAV,EAAmB,KAAhB,C;;IAA ID,C;;MACtC,0D;MAAuB,6BAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;IAAA,C;;MACA,0D;MAAuC,6BAAK,O AAL,EAAc,IAAd,C;MAAvC,Y;IAAA,C;;MACA,0D;MAAwC,6BAAK,SAAL,EAAgB,KAahB,C;MAAxC,Y;IA AA,C;;MAGwG,4BAAiB,OAAjB,EAA0B,KAA1B,C;;IAA1D,C;;MAC9C,kE;MAAuB,qCAAK,IAAL,EAAW,IA AX,C;MAAvB,Y;IAAA,C;;MACA,kE;MAAuC,qCAAK,OAAAL,EAAc,IAAd,C;MAAvC,Y;IAAA,C;;MACA,kE; MAAwC,qCAAK,SAAL,EAAgB,KAahB,C;MAAxC,Y;IAAA,C;;MAGqG,4BAAiB,OAAjB,EAA0B,KAA1B,C;;I AA1D,C;;MAC3C,+D;MAAuB,kCAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;IAAA,C;;MACA,+D;MAAuC,kCAA K,OAAAL,EAAc,IAAd,C;MAAvC,Y;IAAA,C;;MACA,+D;MAAwC,kCAAK,SAAL,EAAgB,KAahB,C;MAAxC,Y ;IAAA,C;;8BAGuG,O;;IAAxD,C;;MAC/C,mE;MAAuB,sCAAK,IAAL,C;MAAvB,Y;IAAA,C;;MAG+G,4BAAiB, OAAjB,EAA0B,KAA1B,C;;IAA1D,C;;MACrD,yE;MAAuB,4CAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;IAAA,C ;MACA,yE;MAAuC,4CAAK,OAAAL,EAAc,IAAd,C;MAAvC,Y;IAAA,C;;MACA,yE;MAAwC,4CAAK,SAAL,EA AgB,KAahB,C;MAAxC,Y;IAAA,C;;MAG6G,4BAAiB,OAAjB,EAA0B,KAA1B,C;;IAA1D,C;;MACnD,uE;MA AuB,0CAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;IAAA,C;;MACA,uE;MAAuC,0CAAK,OAAAL,EAAc,IAAd,C;M AAvC,Y;IAAA,C;;MACA,uE;MAAwC,0CAAK,SAAL,EAAgB,KAahB,C;MAAxC,Y;IAAA,C;;sCAI2G,O;;IAAh E,C;;MAC3C,+D;MAAuB,kCAAK,IAAL,C;MAAvB,Y;IAAA,C;;8BAIkG,O;;IAAxD,C;;MAC1C,8D;MAAuB,i CAAK,IAAL,C;MAAvB,Y;IAAA,C;;8BAGgG,O;;IAAxD,C;;MACxC,4D;MAAuB,+BAAK,IAAL,C;MAAvB,Y;I AAA,C;;MAG+F,mBAAM,OAAN,EAAe,KAaf,C;;IAA3D,C;;MACpC,wD;MAAuB,sBAAK,IAAL,Q;MAAvB,Y ;IAAA,C;;MACA,wD;MAAgC,2BAAK,OAAAL,EAAc,IAAd,C;MAAhC,Y;IAAA,C;;MACA,wD;MAAiD,IAAY,I; MAAzB,2BAAa,SAAR,OAAQ,CAAb,EAAyB,sDAAzB,C;MAApC,Y;IAAA,C;;8BAGoG,O;;IAAxD,C;;MAC5C, gE;MAAuB,mCAAK,IAAL,C;MAAvB,Y;IAAA,C;;8BAliG,O;;IAAxD,C;;MACzC,6D;MAAuB,gCAAK,IAAL,C ;MAAvB,Y;IAAA,C;;MAG4G,4BAAiB,OAAjB,EAA0B,KAA1B,C;;IAA1D,C;;MACID,sE;MAAuB,yCAAK,IA AL,EAAW,IAAX,C;MAAvB,Y;IAAA,C;;MACA,sE;MAAuC,yCAAK,OAAAL,EAAc,IAAd,C;MAAvC,Y;IAAA,C ;MACA,sE;MAAwC,yCAAK,SAAL,EAAgB,KAahB,C;MAAxC,Y;IAAA,C;;MAGoH,4BAAiB,OAAjB,EAA0B, KAA1B,C;;IAA1D,C;;MAC1D,8E;MAAuB,iDAAK,IAAL,EAAW,IAAX,C;MAAvB,Y;IAAA,C;;MACA,8E;MA AuC,iDAAK,OAAAL,EAAc,IAAd,C;MAAvC,Y;IAAA,C;;MACA,8E;MAAwC,iDAAK,SAAL,EAAgB,KAahB,C; MAAxC,Y;IAAA,C;;MCIHA,MAAM,yBAAqB,OAArB,C;IACV,C;;MAII,MAAM,uBAAmB,cAAAnB,C;IACV,C;; MAII,MAAM,6BAAsB,OAAtB,C;IACV,C;;MAII,MAAM,4CAAqC,uBAAqB,YAArB,8BAArC,C;IACV,C;;MC2 BW,kBhGoBiD,oB;MgGO9C,Q;MAAA,OAAK,0B;MAAf,OAAU,cAAV,C;QAAU,mB;QACN,UAAU,sBAAM,C AAN,C;QACV,kBAAkB,sBAAY,GAAZ,C;QakFiD,U;QajFnE,WhGmHJ,agGnHgB,GhGmHhB,EiGxLoB,CDqE kC,uBAAuB,CAAC,WAAY,mBAAY,GAAZ,CAiFhD,GctJrC,CDsJqC,GAA6B,UAjFjC,WaiFiC,6DCtJnD,IAAM

,CAAN,IjGwLpB,C;;MiGxLA,ODuEO,W;ICvEP,C;;;;MCsF0C,OAAI,kBAAK,CAAL,MAAJ,GAA Y,CAAZ,GA
AmB,C;IAAnB,C;;MAOA,OAAI,kBAAK,CAAL,MAAJ,GAA Y,CAAZ,GAAMb,C;IAAnB,C;;MCIDpB,Q;MADt
B,UAAmB,E;MACnB,wBAAsB,KAAAtB,gB;QAAsB,aAAA,KAAAtB,M;QAAK,IAAC,0BAAD,EAAO,2B;QACR,I
AAI,IAAJ,IAAY,K;;MAEhB,OAAO,G;IACX,C;;MAQgB,Q;MADZ,WAA0B,MAAA,MAAK,KAAL,C;MACvC,w
BAAY,IAAZ,gB;QAAY,UAAA,IAAZ,M;QACI,IAAU,KAA Y,gBAAe,GAAf,CAAtB,C;UACI,UAAK,GAAL,IAA
Y,MAAM,GAAN,C;;;MAGpB,OAAO,S;IACX,C;qEC5DA,yB;MAAA,iB;MAAA,oB;QAOkD,OAAA,MAAW,K
AAI,CAAJ,C;MAAX,C;KAPID,C;qEASA,yB;MAAA,iB;MAAA,oB;QAOkD,OAAA,MAAW,KAAI,CAAJ,C;MA
AX,C;KAPID,C;qEASA,yB;MAAA,iB;MAAA,oB;QAOkD,OAAA,MAAW,KAAI,CAAJ,C;MAAX,C;KAPID,C;u
EASA,yB;MAAA,iB;MAAA,oB;QASmD,OAAA,MAAW,MAAK,CAAL,C;MAAX,C;KATnD,C;uEAWA,yB;MA
AA,iB;MAAA,oB;QASmD,OAAA,MAAW,MAAK,CAAL,C;MAAX,C;KATnD,C;uEAWA,yB;MAAA,iB;MAA
A,oB;QASmD,OAAA,MAAW,MAAK,CAAL,C;MAAX,C;KATnD,C;uEAWA,yB;MAAA,iB;MAAA,uB;QAKb+
D,OAAA,MAAW,OAAM,CAAN,EAAS,CAAT,C;MAAX,C;KAIB/D,C;uEAoBA,yB;MAAA,iB;MAAA,oB;QAU
mD,OAAA,MAAW,MAAK,CAAL,C;MAAX,C;KAVnD,C;uEAYA,yB;MAAA,iB;MAAA,oB;QASmD,OAAA,M
AAW,MAAK,CAAL,C;MAAX,C;KATnD,C;uEAWA,yB;MAAA,iB;MAAA,oB;QAUmD,OAAA,MAAW,MAAK
,CAAL,C;MAAX,C;KAVnD,C;yEAYA,yB;MAAA,iB;MAAA,oB;QAYoD,OAAA,MAAW,OAAM,CAAN,C;MA
AX,C;KAZpD,C;yEAcA,yB;MAAA,iB;MAAA,oB;QAYoD,OAAA,MAAW,OAAM,CAAN,C;MAAX,C;KAZpD,
C;yEAcA,yB;MAAA,iB;MAAA,oB;QAaoD,OAAA,MAAW,OAAM,CAAN,C;MAAX,C;KAbpD,C;yEAeA,yB;M
AAA,iB;MAAA,uB;QAS+D,OAAA,MAAW,OAAM,CAAN,EAAS,CAAT,C;MAAX,C;KAT/D,C;uEAWA,yB;M
AAA,iB;MAAA,oB;QAQmD,OAAA,MAAW,MAAK,CAAL,C;MAAX,C;KARnD,C;qEAUA,yB;MAAA,iB;MA
AA,oB;QAUkD,OAAA,MAAW,KAAI,CAAJ,C;MAAX,C;KAVID,C;yEAYA,yB;MAAA,iB;MAAA,oB;QAcOD,
OAAA,MAAW,OAAM,CAAN,C;MAAX,C;KAdpD,C;;MA8BI,IAAI,QAAQ,GAAR,IAAe,SAAQ,GAA3B,C;QA
AgC,OAAO,wCAA O,I;MAC9C,OAAO,IAAW,KAAI,CAAJ,CAAX,GAAoB,IAAW,KAAI,IAAJ,C;IAC1C,C;mE
AEA,yB;MAAA,iB;MAAA,oB;QAWiD,OAAA,MAAW,KAAI,CAAJ,C;MAAX,C;KAXjD,C;yEAaA,yB;MAAA,i
B;MAAA,oB;QAOoD,OAAA,MAAW,OAAM,CAAN,C;MAAX,C;KAPpD,C;uEASA,yB;MAAA,iB;MAAA,oB;
QAOmD,OAAA,MAAW,MAAK,CAAL,C;MAAX,C;KAPnD,C;uEASA,yB;MAAA,iB;MAAA,oB;QAgBmD,OA
AA,MAAW,OAAM,CAAN,C;MAAX,C;KAhBnD,C;uEAkBA,yB;MAAA,iB;MAAA,oB;QAUmD,OAAA,MAA
W,MAAK,CAAL,C;MAAX,C;KAVnD,C;yEAYA,yB;MAAA,iB;MAAA,oB;QAUoD,OAAA,MAAW,OAAM,CA
AN,C;MAAX,C;KAVpD,C;+EAYA,yB;MAAA,iB;MAAA,oB;QAUuD,OAAA,MAAW,OAAM,CAAN,C;MAAX,
C;KAVvD,C;;MAoBI,IAAI,IAAI,GA AJ,KAAW,GAAf,C;QACI,OAAO,IAAW,OAAM,CAAN,C;;MAEtB,YAzBg
D,MAAW,OAYBzC,CAzByC,C;MA0B3D,OAAW,QAAQ,CAAR,KAAa,GAAxB,GAA6B,KAA7B,GAtC+C,MA
AW,MA sCb,CAtCa,C;IAuC9D,C;qEAEA,yB;MAAA,iB;MAAA,oB;QAUkD,OAAA,MAAW,KAAI,CAAJ,C;MA
AX,C;KAVID,C;uEAYA,yB;MAAA,iB;MAAA,oB;QAWmD,OAAA,MAAW,MAAK,CAAL,C;MAAX,C;KAXn
D,C;wEAcA,yB;MAAA,iB;MAAA,uB;QAO6D,OAAA,MAAW,KAAI,CAAJ,EAAO,CAAP,C;MAAX,C;KAP7D,
C;wEASA,yB;MAAA,iB;MAAA,uB;QAO6D,OAAA,MAAW,KAAI,CAAJ,EAAO,CAAP,C;MAAX,C;KAP7D,C;
qEAWA,yB;MAAA,iB;MAAA,+B;QAayD,OAAA,MAAW,KAAI,SAAJ,EAAU,CAAV,C;MAAX,C;KAbzD,C;uE
AeA,yB;MAAA,iB;MAAA,+B;QAOsD,OAAA,MAAW,KAAI,SAAJ,EAA Y,CAAZ,C;MAAX,C;KAPtD,C;iGAm
BsD,yB;MAAA,iB;MAAA,4B;QAAQ,OAAA,MAAW,KAAI,SAAJ,C;MAAX,C;KAAR,C;+EAaT,yB;MAAA,iB;
MAAA,4B;QAAQ,OAAA,MAAW,MAAK,SAAL,C;MAAX,C;KAAR,C;IFAE7C,yB;MAAA,6C;MAAA,kC;QAK
8D,OAAK,SAAL,SAAK,EAAC,IAAd,C;MAAL,C;KAL9D,C;;MAmBI,gBAAO,CAAP,C;QADyC,OACrB,QAAP,
CAAC,SAAM,C;WACpB,IAAK,QAAL,SAAK,CAAL,IAAgB,cAAQ,wCAA O,kBAA/B,C;QAFyC,OA EW,S;WA
CpD,kBAAQ,wCAA O,UAAf,C;QAHyC,OAGb,YAAY,SAAL,SAAK,C;;QAHc,OAI5B,OAAL,SAAK,CAAL,GA
AgB,S;IAJiB,C;;MAYzC,IAAK,QAAL,SAAK,CAAL,IAAgB,cAAQ,wCAA O,kBAA/B,C;QADwC,OACY,S;WA
CpD,kBAAQ,GAAR,C;QAFwC,OA EZB,wCAA O,U;;QACP,WAAc,UAA L,SAAK,CAAL,yBAAuB,YAAO,CAA
X,GAAc,CAAd,GAAqB,EAAX,C,E;QAHgB,O3Cjb6B,MAAA,gBAAe,IAAf,C;;I2Cib1C,C;;MAWxC,IAAK,QAAL
,SAAK,CAAL,IAAgB,cAAQ,wCAA O,kBAA/B,C;QAD0C,OACU,S;WACpD,kBAAQ,GAAR,C;QAF0C,OAE3B,
CAAC,wCAA O,U;;QACR,WAAc,UAA L,SAAK,CAAL,yBAAuB,YAAO,CAAX,GAAc,EAAd,GAA sB,CAAzC,E
;QAHkB,O3C3b2B,MAAA,gBAAe,IAAf,C;;I2C2bxC,C;;MAiB1C,IAAK,QAAL,SAAK,CAAL,IAAmB,QA AH,E
AAG,CAAnB,C;QADuD,OACzB,wCAA O,I;WACrC,WAAM,SAAN,C;QAFuD,OA EZC,E;WACd,SAAK,SAAL,

C;QAHuD,OAGrC,OAAL,SAAK,C;;QAHqC,OAI1B,SAAL,SAAK,C;IAJ0B,C;;MAoBvD,uB;QAAW,MAAM,gC
AAyB,yBAaZB,C;WACjB,gBAAO,UAAp,C;QAFyC,OAeJB,U;WACxB,gBAAO,WAAP,C;QAHyC,OAGjB,W;;
QAHiB,OAIW,YAA5C,IAAW,OAAM,SAAN,CAAiC,C;;IAJX,C;;MAmBzC,uB;QAAW,MAAM,gCAAyB,yBAA
zB,C;WACjB,oD;QAF2C,+B;WAG3C,oD;QAH2C,+B;;QAAA,OAIS,uBAA5C,IAAW,OAAM,SAAN,CAAiC,C;;I
AJT,C;uEAa/C,yB;MAAA,iB;MAAA,oB;QAOgD,OAAA,MAA6B,KAAZ,CAAY,C;MAA7B,C;KAPhD,C;uEAS
A,yB;MAAA,iB;MAAA,oB;QAOgD,OAAA,MAA6B,KAAZ,CAAY,C;MAA7B,C;KAPhD,C;uEASA,yB;MAAA,i
B;MAAA,oB;QAOgD,OAAA,MAA6B,KAAZ,CAAY,C;MAA7B,C;KAPhD,C;yEASA,yB;MAAA,iB;MAAA,oB;
QASiD,OAAA,MAA8B,MAAZ,CAAY,C;MAA9B,C;KATjD,C;yEAWA,yB;MAAA,iB;MAAA,oB;QASiD,OAA
A,MAA8B,MAAZ,CAAY,C;MAA9B,C;KATjD,C;yEAWA,yB;MAAA,iB;MAAA,oB;QASiD,OAAA,MAA8B,M
AAZ,CAAY,C;MAA9B,C;KATjD,C;2EAWA,yB;MAAA,iB;MAAA,uB;QakB4D,OAAA,MAA6C,OAA1B,CAA
0B,EAAZ,CAAY,C;MAA7C,C;KAIB5D,C;yEAoBA,yB;MAAA,iB;MAAA,oB;QAUiD,OAAA,MAA8B,MAAZ,C
AAY,C;MAA9B,C;KAVjD,C;yEAYA,yB;MAAA,iB;MAAA,oB;QASiD,OAAA,MAA8B,MAAZ,CAAY,C;MAA
9B,C;KATjD,C;yEAWA,yB;MAAA,iB;MAAA,oB;QAUiD,OAAA,MAA8B,MAAZ,CAAY,C;MAA9B,C;KAVjD,
C;2EAYA,yB;MAAA,iB;MAAA,oB;QAYkD,OAAA,MAA+B,OAAZ,CAAY,C;MAA/B,C;KAZID,C;2EAca,yB;
MAAA,iB;MAAA,oB;QAYkD,OAAA,MAA+B,OAAZ,CAAY,C;MAA/B,C;KAZID,C;2EAca,yB;MAAA,iB;MA
AA,oB;QAakD,OAAA,MAA+B,OAAZ,CAAY,C;MAA/B,C;KAbID,C;2EAeA,yB;MAAA,iB;MAAA,uB;QAS4D,
OAAA,MAA6C,OAA1B,CAA0B,EAAZ,CAAY,C;MAA7C,C;KAT5D,C;yEAWA,yB;MAAA,iB;MAAA,oB;QAQ
iD,OAAA,MAA8B,MAAZ,CAAY,C;MAA9B,C;KARjD,C;uEAUA,yB;MAAA,iB;MAAA,oB;QAUgD,OAAA,M
AA6B,KAAZ,CAAY,C;MAA7B,C;KAVhD,C;2EAYA,yB;MAAA,iB;MAAA,oB;QackD,OAAA,MAA+B,OAAZ,
CAAY,C;MAA/B,C;KAdID,C;uEAgBA,yB;MAAA,mC;MAAA,0B;Qac6D,OAAmC,IAA7B,CAA6B,EAAZ,IAA
Y,C;MAAnC,C;KAd7D,C;qEAgBA,yB;MAAA,iB;MAAA,oB;QAW+C,OAAA,MAA6B,KAAZ,CAAY,C;MAA7
B,C;KAX/C,C;2EAaA,yB;MAAA,iB;MAAA,oB;QAOkD,OAAA,MAA+B,OAAZ,CAAY,C;MAA/B,C;KAPID,C;
yEASA,yB;MAAA,iB;MAAA,oB;QAOiD,OAAA,MAA8B,MAAZ,CAAY,C;MAA9B,C;KAPjD,C;yEASA,yB;M
AAA,iB;MAAA,oB;QAgBiD,OAAA,MAA+B,OAAZ,CAAY,C;MAA/B,C;KAhBjD,C;yEakBA,yB;MAAA,iB;M
AAA,oB;QAUiD,OAAA,MAA8B,MAAZ,CAAY,C;MAA9B,C;KAVjD,C;2EAYA,yB;MAAA,iB;MAAA,oB;QA
UkD,OAAA,MAA+B,OAAZ,CAAY,C;MAA/B,C;KAVID,C;iFAYA,yB;MA3gBA,iB;MA2gBA,oB;QAUqD,OA3
gBE,MAAW,OA2gBF,CA3gBE,C;MA2gBb,C;KAVrD,C;2EAYA,yB;MAAA,uC;MAAA,oB;QAQkD,OAAoB,M
AAZ,CAAY,C;MAApB,C;KARID,C;uEAWA,yB;MAAA,iB;MAAA,oB;QAUgD,OAAA,MAA6B,KAAZ,CAAY,
C;MAA7B,C;KAVhD,C;yEAYA,yB;MAAA,iB;MAAA,oB;QAWiD,OAAA,MAA8B,MAAZ,CAAY,C;MAA9B,C
;KAXjD,C;wEAeA,yB;MAAA,iB;MAAA,uB;QAO0D,OAAA,MAAW,KAAI,CAAJ,EAAO,CAAP,C;MAAX,C;K
APID,C;wEASA,yB;MAAA,iB;MAAA,uB;QAO0D,OAAA,MAAW,KAAI,CAAJ,EAAO,CAAP,C;MAAX,C;KA
PID,C;sEAYA,yB;MAAA,iB;MAAA,+B;QAasD,OAAA,MAA8C,KAA1B,SAA0B,EAAZ,CAAY,C;MAA9C,C;K
AbtD,C;uEAeA,yB;MAAA,iB;MAAA,+B;QAOoD,OAAA,MAA8C,KAA1B,SAA0B,EAAZ,CAAY,C;MAA9C,C;
KAPpD,C;kGAmBoD,yB;MAAA,iB;MAAA,4B;QAAQ,OAAA,MAAgC,KAAZ,SAAY,C;MAAhC,C;KAAR,C;gF
AaT,yB;MAAA,iB;MAAA,4B;QAAQ,OAAA,MAAiC,MAAZ,SAAY,C;MAAjC,C;KAAR,C;gFAE3C,yB;MAAA,
6C;MAAA,kC;QAO8D,OAA0C,SAArC,SAAqC,EAAZ,IAAY,C;MAA1C,C;KAP9D,C;iFASA,yB;MAAA,6C;MA
AA,kC;QAK4D,OAA0C,SAArC,SAAqC,EAAZ,IAAY,C;MAA1C,C;KAL5D,C;oFAQA,yB;MAAA,iD;MAAA,4B
;QAYmD,OAAW,WAAX,SAAW,C;MAAX,C;KAZnD,C;sFAca,yB;MAAA,mD;MAAA,4B;QAYqD,OAAW,YA
AX,SAAW,C;MAAX,C;KAZrD,C;;MA8BqC,OAAI,IAAI,CAAR,GAAY,CAAC,CAAD,OAAM,CAAIB,GAA0B,
C;IAA1B,C;wEAerC,yB;MAAA,iB;MAAA,uB;QAKoD,OAAA,MAAW,KAAI,CAAJ,EAAO,CAAP,C;MAAX,C;
KALpD,C;wEAOA,yB;MAAA,iB;MAAA,uB;QAKoD,OAAA,MAAW,KAAI,CAAJ,EAAO,CAAP,C;MAAX,C;K
ALpD,C;mGaiBgD,yB;MAAA,mC;MAAA,4B;QAAQ,WAAI,SAAJ,C;MAAA,C;KAAR,C;;MAU5C,gBAAO,CA
AP,C;QADoC,OACxB,E;WACZ,gBAAO,CAAP,C;QAFoC,OAExB,C;;QAFwB,OAG5B,C;IAH4B,C;;MAiBD,OA
AI,eAAI,CAAR,GAAY,CAAD,aAAX,GAAMB,C;IAAnB,C;;MAOgB,OAAI,kBAAK,CAAL,MAAJ,GAAY,CAA
Z,GAAMB,C;IAAnB,C;;MAOA,OAAI,kBAAK,CAAL,MAAJ,GAAY,CAAZ,GAAMB,C;IAAnB,C;mGAYL,yB;
MAAA,mC;MAAA,4B;QAAQ,WAAI,SAAJ,C;MAAA,C;KAAR,C;;MAU9C,2BAAO,CAAP,C;QADqC,OACzB,
E;WACZ,2BAAO,CAAP,C;QAFqC,OAezB,C;;QAFyB,OAG7B,C;IAH6B,C;;MCPkCG,qBAAQ,S;IAAR,C;;MA
MD,qBAAQ,S;IAAR,C;;MAKM,qBAAQ,wCAAO,kBAAf,IAAoC,cAAQ,wCAAO,kB;IAAnD,C;;MAKD,qBAAQ

,uCAAM,kBAAd,IAAmC,cAAQ,uCAAM,kB;IAAjD,C;;MAKD,QAAC,qBAAD,IAAiB,CAAC,kB;IAAIB,C;;MAKD,QAAC,uBAAD,IAAiB,CAAC,kB;IAAIB,C;;MAU1C,QAAQ,S;MACR,IAAI,CAAC,IAAM,UAAP,KAAAsB,CAAE,KAAK,CAAP,GAAC,UAApC,K;MACJ,IAAI,CAAC,IAAM,SAAP,KAAAsB,CAAE,KAAK,CAAP,GAAC,SAApC,K;MACJ,IAAI,CAAC,IAAM,SAAP,KAAAsB,CAAE,KAAK,CAAP,GAAC,SAApC,K;MACJ,IAAI,CAAC,IAAM,QAAP,KAAAsB,CAAE,KAAK,CAAP,GAAC,QAApC,K;MACJ,IAAI,CAAC,IAAM,KAAP,KAAAsB,CAAE,KAAK,EAA7B,K;MACJ,OAAO,C;IACX,C;kGAEA,yB;MAAA,iB;MAAA,4B;QAO2D,OAAe,aAAM,SAAN,C;MAAf,C;KAP3D,C;;MAGBI,YATsE,aAStC,EAAf,aAAQ,CAAC,SAAD,IAAR,CAAe,CATsC,CAStE,I;IAAA,C;;MASA,OAAI,cAAQ,CAAZ,GAAe,CAAf,GAAsB,CAAE,IAAI,EAAJ,GAIB8C,uB;IAkBE,C;;MAUA,oBAAS,CAAC,SAAD,IAAT,C;IAAA,C;;MAeA,oBAAI,QAAJ,GAAiB,cAAK,EAAL,GAAqB,Q;IAAtC,C;;MAGBA,oBAAI,EA AJ,GAAoB,QAApB,GAAiC,cAAK,Q;IAAtC,C;;MASA,OAAK,a5CIEmD,uB4CkEnD,CAAL,GAA0B,a5CnE6B,sB4CmE7B,CAA1B,I;IAAA,C;;MAQM,WAAW,S5C1EuC,c;M4C2EpD,e;QADJ,OACS,KA7E6D,a5CCf,sB4CDe,CA6E7D,I;;QADT,OA5EsE,aA8E1D,IA9E0D,C;;IA4EtE,C;;MAWM,UAAU,S5CtFuC,a;M4CuFnD,c;QADJ,OA CS,KAAqB,sB5CtF0B,uB4CsF1B,CAArB,I;;QADT,OAEGb,sBAAJ,GAAI,C;IAFhB,C;;MAYM,WAAW,S5CjGu C,c;M4CkGpD,e;QAAK,UAAAS,kB5CnGqC,sB4CmGrC,C;QADIB,O5CnG4C,MAAA,KAAK,UAAAS,GAAT,E4Co GvB,C5CpGuB,C;;Q4CqGID,aAAa,kBAAL,IAAK,C;QAFzB,O5CnG4C,MAAA,KAAK,U4CqG7C,C5CrG6C,EA Ac,MAAd,C;;I4CmG9D,C;;MAYM,UAAU,S5C9GuC,a;M4C+GnD,c;QAAK,WAAa,iB5C9GkC,uB4C8GIC,C;QA DtB,O5C/G4C,MAAA,KAAK,U4CgHhD,C5ChHgD,EAAC,IAAd,C;;Q4CiHID,YAAS,iBAAJ,GAAI,C;QAFrB,O5 C/G4C,MAAA,KAAK,UAAAS,KAAT,E4CiHrB,C5CjHqB,C;;I4C+G9D,C;;MAkBA,IAAI,CAAC,WAAa,EAAd,MA AqB,CAAzB,C;QACI,UAAU,S5CjIyC,a;Q4CkInD,WAAW,S5CjIyC,c;Q4CkIpD,aAAa,GAAL,IAAI,QAAR,GA AqB,IAAK,MAAK,CAAC,QAAD,IAAL,C;QACvC,cAAc,IAAK,IAAI,QAAT,GAAsB,GAAI,MAAK,CAAC,QA AD,IAAL,C;QACxC,OAAW,CAAC,WAAa,EAAd,MAAqB,CAAhC,G5CtIwC,MAAA,KAAK,U4CsIIB,M5CtIkB, E4CsIV,O5CtIU,C4CsIID,G5CtIwC,MAAA,KAAK,U4CsIS,O5CtIT,E4CsIkB,M5CtIIB,C;;Q4CwInD,Q;QAAA,I AAI,CAAC,WAAa,EAAd,MAAqB,CAAzB,C;UAAA,OAA4B,S;;uB5CtIiB,uB;U4CsIP,a5CvIM,sB;U4CuI5C,O5 CxIiC,MAAA,KAAK,kBAAc,MAAd,C;;Q4CwI1D,W;;IAER,C;kFAGA,yB;MAAA,4C;MAAA,sC;QAaiE,6BAA W,CAAC,QAAD,IAAX,C;MAAA,C;KAbjE,C;;MCtKI,OAAO,SAA8B,MAAK,WAAL,C;IACzC,C;;MAMI,OAA O,SAA8B,MAAK,WAAL,EAakB,UAAIB,C;IACzC,C;;MCpCI,gBAAH,IAAI,OAAO,EAAG,GAAE,IAAI,IAAI, CAAC,CAAD,EAAI,EAAJ,CAAd,GAAyB,CAAhC,C;IAAA,C;;MAMA,cAAO,QAAP,GAakB,QAAQ,Q;IAA1 B,C;8EtFTJ,yB;MAAA,6C;MAAA,kC;QAGI,OAAgB,QAAx,SAAW,EAAa,IAAb,C;MAAhB,C;KAHJ,C;;MuFm BY,QxDdK,OwDcnB,SAAf,C;aACJ,Q;UADI,OACQ,M;aACZ,Q;UAFI,OAEQ,M;aACZ,S;UAHI,OAGS,O;gBA HT,OAI,MAAA,gBA Ae,SAAf,CAAqB,Y;;IAJtC,C;;MAWC,Q;MAAD,OAAwB,CAAvB,iEAAuB,Q;IAAxB,C;;M AMA,iBAAU,SAAV,C;IAAA,C;;MCjCR,8B;IADuC,C;;MACvC,4B;IAAA,C;;M/FWgC,MAAM,yB;I+FR1B,C;; M/FQoB,MAAM,yB;I+FN1B,C;;M/FMoB,MAAM,yB;I+FJ1B,C;;M/FIoB,MAAM,yB;I+FF1B,C;;M/FEoB,MAA M,yB;I+FA1B,C;;M/FAoB,MAAM,yB;I+FE1B,C;;M/FFoB,MAAM,yB;I+FI1B,C;;M/FJoB,MAAM,yB;I+FM1B,C ;M/FNoB,MAAM,yB;I+FQ1B,C;;M/FRoB,MAAM,yB;I+FU1B,C;;M/FVoB,MAAM,yB;I+FY1B,C;;M/FZoB,MA AM,yB;I+Fc1B,C;;M/FdoB,MAAM,yB;I+FgB1B,C;;M/FhBoB,MAAM,yB;I+FkB1B,C;;M/FIBoB,MAAM,yB;I+F oB1B,C;;M/FpBoB,MAAM,yB;I+Fsb1B,C;;M/FtBoB,MAAM,yB;I+FwB1B,C;;MAGR,OAAO,oCAA0B,oBAAU, KAAM,OAAhB,C;IACrC,C;;MAG+B,gB;MAAA,8FAA0B,C;IAA1B,C;;MAI3B,OAAO,oBAAQ,eAAR,C;IACX, C;;MAG2D,sBAAc,MAAd,C;MACL,Q;MAAtD,4BAAmC,CAAmB,OAAZ,MAAY,WAAAnB,kC;IADC,C;;MACp C,gC;IAAA,C;;MAGW,cAAgB,W;MAAvB,OxDqCuD,MAAA,QwDrCpD,KxDqCoD,EAAy,OAAZ,C;IwDpCxE, C;;MAOA,sBAAc,MAAd,C;MAFA,wC;MACA,8C;IAHuC,C;;MAMnC,IAAI,0CAAJ,C;QAAsC,OAAO,K;MAC 7C,OAAa,uCAAQ,KAAP,CAAN,IAAuB,+BAAmB,KAAM,kBAAzB,C;IACIC,C;;MAEyC,6B;IAAA,C;;MAGrC, OAAO,0BAAmB,KAAAnB,C;IACX,C;;MAGJ,iC;MAAoC,sBAAoB,MAApB,C;MACHc,4BAAkC,S;IADtC,C;;M ACI,gC;IAAA,C;;MAEGD,Y;IAAA,C;;MAGpC,MAAM,qCAA8B,6CAA9B,C;IAAN,C;;MAEGc,iBAAU,I;IAAV, C;;MAEb,Q;IAAA,C;;MAVnC,wC;QAAA,uB;;MAAA,iC;IAAA,C;;MC5EA,gC;MAEI,gBACe,wBAAoB,MAAp B,EAA6D,KAA7D,EAAoE,gCAApE,C;MAEf,mBACkB,wBAAoB,MAApB,EAAGe,QAAhE,EAA0E,mCAA1E,C ;MAEIB,oBACmB,+B;MAEnB,oBACmB,wBAAoB,OAApB,EAake,SAaIE,EAA6E,oCAA7E,C;MAEnB,iBACg B,wBAAoB,MAApB,EAA8D,MAA9D,EAASe,iCAAtE,C;MAEhB,kBACiB,wBAAoB,MAApB,EAA+D,OAA/D, EAAwE,kCAAxE,C;MAEjB,gBACe,wBAAoB,MAApB,EAA6D,KAA7D,EAAoE,gCAApE,C;MAEf,kBACiB,wB

AAoB,MAApB,EAA+D,OAA/D,EAAwE,kCAAxE,C;MAEjB,mBACkB,wBAAoB,MAApB,EAAgE,QAAhE,EA
A0E,mCAA1E,C;MAEIB,kBACiB,wBAAoB,KAApB,EAAiE,OAAjE,EAA0E,kCAA1E,C;MAEjB,mBACkB,wB
AAoB,MAApB,EAAgE,QAAhE,EAA0E,mCAA1E,C;MAEIB,sBACqB,wBAAoB,KAApB,EAAkE,WAAIE,EAA
+E,sCAA/E,C;MAErB,yBACwB,wBAAoB,KAApB,EAAqE,cAArE,EAAqF,yCAArF,C;MAExB,sBACqB,wBAA
oB,WAApB,EAAwE,WAAxE,EAAqF,sCAArF,C;MAErB,sBACqB,wBAAoB,SAApB,EAAeE,WAAtE,EAAmF,s
CAAnF,C;MAErB,uBACsB,wBAAoB,UAApB,EAAwE,YAAxE,EAAfE,uCAAtF,C;MAEtB,qBACoB,wBAAoB,
UAApB,EAAeE,UAAtE,EAAkF,qCAAIF,C;MAEpB,sBACqB,wBAAoB,KAApB,EAAkE,WAAIE,EAA+E,sCAA/
E,C;MAErB,uBACsB,wBAAoB,YAApB,EAA0E,YAA1E,EAAwF,uCAAxF,C;MAEtB,wBACuB,wBAAoB,YAA
pB,EAA2E,aAA3E,EAA0F,wCAA1F,C;IA5D3B,C;;MAkE6C,qB;QAAE,O1D/DD,O0D+DU,EAAT,KAAiB,UAA
jB,IAAkC,EAAY,OAAf,KAA0B,a;MAA3D,C;IAAA,C;;MAFd,Q;MAAhB,U;MAAA,KAAgB,OAAhB,eAAgB,C
AAI,KAAJ,CAAhB,U;QAAA,a;;QACH,aAAa,wBAAoB,QAApB,EAA+D,kBAA/D,EACoB,mDADpB,C;QAEg,
AAhB,UAAqC,M;QAHIC,SAIH,M;;MAJ,J,a;IAMJ,C;;MAnEqF,6B;IAAU,C;;MAGD,0B;IAAa,C;;MAMT,8B;IAA
c,C;;MAGxB,6B;IAAW,C;;MAGR,6B;IAAY,C;;MAGiB,6B;IAAU,C;;MAGJ,6B;IAAY,C;;MAGT,6B;IAAa,C;;M
AGd,yB;IAAe,C;;MAGd,6B;IAAa,C;;MAGL,mC;IAAgB,C;;MAGP,gC;IAAmB,C;;MAGtB,6B;IAAgB,C;;MAGl
B,6B;IAAgB,C;;MAGZ,8B;IAAiB,C;;MAGvB,4B;IAAe,C;;MAGjB,6B;IAAgB,C;;MAGN,8B;IAAiB,C;;MAGd,+
B;IAAkB,C;;MA5DzI,uC;QAAA,sB;;MAAA,gC;IAAA,C;;MCEkE,yBAaKB,MAAiB,C;IAAA,C;;MAIxD,IAA
N,I;MAAA,Q3DH0C,O2DG3B,CAAf,C;aACI,Q;UAA6B,OAAjB,8BAAiB,Y;UAA7B,K;aACA,Q;UAAy,OAAI,
CAAY,CpDAhC,GoDAmC,CAAf,MAAkC,CAAtC,GAAyC,8BAAiB,SAAI,D,GAAwE,8BAAiB,Y;UAArG,K;aA
CA,S;UAA8B,OAAjB,8BAAiB,a;UAA9B,K;aACA,U;UAA+B,OAAjB,8BAAiB,eAAgB,CAAY,OAA5B,C;UAA/
B,K;gBAGQ,6B;YAAc,OAAjB,8BAAiB,kB;eACtC,0B;YAAmC,OAAjB,8BAAiB,e;eACnC,0B;YAAmC,OAAj
B,8BAAiB,e;eACnC,2B;YAAoC,OAAjB,8BAAiB,gB;eACpC,yB;YAAkC,OAAjB,8BAAiB,c;eAClC,0B;YAAmC
,OAAjB,8BAAiB,e;eACnC,2B;YAAoC,OAAjB,8BAAiB,gB;eACpC,4B;YAAqC,OAAjB,8BAAiB,iB;eACrC,6B;;
eACA,sB;YAAkC,OAAjB,8BAAiB,W;;YAE9B,kBAaKB,MAAA,gBAaE,CAAf,CAaKB,Y;YAE7C,oBAaKB,MA
AhB,C;cAAiD,OAAjB,8BAAiB,S;iBACjD,oBAaKB,KAAhB,C;cAAgD,OAAjB,8BAAiB,e;;cAE5C,cAA0B,W;cA
C1B,yBAaKB,OAAIB,C;;UAXBxB,K;;MAAA,W;IAAA,C;;MAqCO,Q;MAJP,IAAI,WAAW,MAAf,C;QAA6B,O
AAO,8BAAiB,Y;;MAErD,eAAsB,MAAY,W;MAE3B,IAAI,gBAAJ,C;QACH,IAAI,QAAS,SAAT,QAAJ,C;UACI,
aAAa,qBAAiB,MAAjB,C;UACb,oBAAsB,M;UACtB,a;;UAES,OAAT,QAAS,S;;QAGb,4BAAiB,MAAjB,C;;M
ATJ,W;IAWJ,C;;MC9BI,sBAAY,C;IACHB,C;;MAiBkE,iBAAY,KAAZ,C;IAAA,C;;MAMJ,gB;IAAA,C;;MCjD1D
,qBAAwC,Q;IADK,C;;MAI1B,Q;MAAA,yB;MAAA,iB;QAAe,MAAM,6BAAsB,0CAAtB,C;;MAApC,eAAe,I;M
ACf,qBAAc,I;MACd,OAAO,QAAS,W;IACpB,C;;MCN6C,OAAW,QAAX,8BAAW,EAAQ,YAAR,C;IAAX,C;qF
AEjD,yB;MAAA,0B;MAAA,4B;QACoD,OAAiF,OAArI,MAAoD,cAA0B,aAA1B,CAAmC,cAAc,YAAW,CAAX,
CAAgC,C;MAAJF,C;KADpD,C;qFAGA,yB;MAAA,0B;MAAA,4B;QACoD,OAAiF,OAArI,MAAoD,cAA0B,aAA
1B,CAAmC,cAAc,YAAW,CAAX,CAAgC,C;MAAJF,C;KADpD,C;;MAMoD,QAAQ,cAAA,sCAAk,mBAAL,EA
AyB,sCAAk,mBAA9B,CAAR,6B;IAAA,C;;MAKD,QAAQ,cAAA,sCAAk,kBAAL,EAAwB,sCAAk,kBAA7B,C
AAR,6B;IAAA,C;;MCbH,cxFyKoB,uBwFzKpB,EAAiB,MAAjB,C;IAAA,C;;MAMN,Q;MAAA,yDAaKB,kBAaK
B,SAAiB,C;IAAiB,C;;MAOU,Q;MAAA,yCAAA,KAAb,oBAAuB,kBAaKB,SAAiB,C;IAAvB,C;;MAOR,Q;MAA
A,0DAAmB,kBAaKB,SAAiB,C;IAAnB,C;;MAOU,Q;MAAA,0CAAc,KAAd,oBAAwB,kBAaKB,SAAiB,C;IAAx
B,C;;MAMd,Q;MAAA,wDAAiB,kBAaKB,SAAiB,C;IAAjB,C;;MAOU,Q;MAAA,wCAAY,KAAZ,oBAAsB,kBA
AkB,SAAiB,C;IAAtB,C;;MAMR,Q;MAAA,yDAaKB,kBAaKB,SAAiB,C;IAAiB,C;;MAOU,Q;MAAA,yCAAA,K
AAb,oBAAuB,kBAaKB,SAAiB,C;IAAvB,C;;MAMqC,kBAA1C,CAAO,S;MACID,IAAO,QrG0BD,WqG1BC,CA
AH,IAAc,CAAM,kBAApB,KrG0BE,WqG1B6B,KAAM,GAAN,IAAkB,kBAAjD,CAAJ,C;QACI,4B;MAFsC,OrG
4BnC,W;IqG5BmC,C;6EAK9C,yB;MAAA,6C;MAAA,4B;QAKmD,0B;MAAA,C;KALnD,C;;MAWgG,kBAA1C,
CAAO,S;MAAR,OACjD,EAAK,QrGoCgB,WqGpChB,CAAH,IAAc,CAAM,kBAApB,KrGoCmB,WqGpCY,KAA
M,GAAN,IAAkB,kBAAjD,CAAF,CrGoCO,GAAqB,WAArB,GAA+B,I;IqGrCW,C;yFAIrD,yB;MAAA,yD;MAA
A,4B;QAK0D,gC;MAAA,C;KAL1D,C;iFAOA,yB;MAAA,6C;MAAA,mC;QAO6D,OAAa,SAAR,SAAQ,EAAS,K
AAT,C;MAAb,C;KAP7D,C;iFASA,yB;MAAA,6C;MAAA,mC;QAO8D,OAAa,SAAR,SAAQ,EAAS,KAAT,C;M
AAb,C;KAP9D,C;;MAeqD,OAAA,SAAY,UAAS,WAAW,KAAX,CAAT,C;IAAZ,C;;MAEf,QAAAM,SxFyDwB,cw
FzD9B,C;aACIC,K;aAAA,M;aAAA,M;UADkC,OACT,I;gBADS,OAE1B,K;;IAF0B,C;;MAUIC,IAAI,EAAU,CAA

V,sBAaA,EAAb,CAAJ,C;QACI,MAAM,gCAAYB,WAAQ,KAAR,kCAAzB,C;;MAEV,OAAO,K;IACX,C;;MAE2
D,Q;MACvD,YAAQ,EAAR,IAAe,QAAQ,EAAvB,C;QAA8B,cAAO,E;WACrC,YAAQ,EAAR,IAAe,QAAQ,EA
vB,C;QAA8B,cAAO,EAAP,GAAa,EAAb,I;WAC9B,YAAQ,EAAR,IAAe,QAAQ,GAAvB,C;QAA8B,cAAO,EA
P,GAAa,EAAb,I;;QACtB,S;MrGhCD,a;MqG4BgD,OAK/C,WAAJ,GAAiB,EAAbJ,GAAyB,E;IAL0B,C;;MCI3D,
e;MAGqC,kB;MAHrC,iB;MAAA,uB;IAGoC,C;;MAHpC,qC;MAAA,C;MAII,qEACY,GADZ,C;MAEA,iEAIU,G
AJV,C;IANJ,C;;MAII,wB;MAAA,uC;IAAA,C;;MAEA,wB;MAAA,qC;IAAA,C;;MANJ,mF;IAAA,C;;MAAA,a
;aAA,a;UAAA,4C;aAA,W;UAAA,0C;gBAAA,4D;;IAAA,C;;MAMBoC,kB;IAAD,C;;MAAC,iB;IALpC,C;;M
AAA,sBAKoC,qCALpC,C;IAAA,C;MAAA,OAKoC,iDALpC,M;IAAA,C;MAAA,c;MAKoC,sD;MALpC,a;IAA
A,C;MAAA,2IAKoC,sCALpC,G;IAAA,C;MAoJI,6B;MA1HA,eACoC,O;MACpC,eACsD,QAAR,OAAQ,C;MpG
4uC/C,kBAAM,iBAaA,wBoG3uC0B,OpG2uC1B,EAawB,EAaxB,CAAb,C;MAqEA,Q;MAAA,OoGhzCuC,OpG
gzCvC,W;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,WAAy,WAAc,IoGjzCuC,MpGizCrD,C;;MoGjzChB,uBAaO,C,
WAAO,OAAP,EAAYC,epGkzCtE,WoGlzCsE,EAAYB,EAazB,CAAzB,GAAwD,GAAxE,C;IAbd,C;;MAiBJ,MAA
d,oBAAc,C;MACd,YAAy,oBAAc,MAAK,KAAM,WAAx,C;MAC1B,OAAO,iBAaiB,KAAM,MAAN,KAAe,CA
AhC,IAAqC,oBAAc,UAAAd,KAA2B,KAAM,O;IACjF,C;;MAIkB,MAAd,oBAAc,C;MACd,OAAO,oBAAc,MAAK
,KAAM,WAAx,C;IACzB,C;;MAQ4C,0B;QAAA,aAAkB,C;MAAmB,OAAC,SAAd,oBAAc,EAAS,KAAM,WAAf
,EAA2B,UAA3B,C;IAAd,C;;MAS5D,mB;QAAE,+BAAK,aAAL,EAAY,kBAAZ,C;MAAF,C;IAAA,C;;MAAwC,
OAAA,KAAM,O;IAAO,C;;MAD3B,0B;QAAA,aAAkB,C;MAC7D,0BAaiB,6CAAjB,EAA8C,sBAA9C,C;IAAA,
C;;MAQA,IAAY,aAAR,YAAQ,EAaw,EAAX,CAAR,IAAmC,WAAR,YAAQ,EAAS,EAAT,CAAvC,C;QACI,O
AAO,kBAAK,KAAL,C;;QAEP,OAAkE,CAA3D,UAAm,MAA2B,UAAf,YAAR,YAAQ,qBAAU,EAav,EAae,qB
AAQ,EAAR,EAA3B,MAAN,EAakD,YAAID,CAA2D,eAAK,KAAL,C;IAC1E,C;;MAO8E,OAAA,KAAM,WzF4
Ga,SyF5GY,oBzF4GZ,EyF5G2B,WzF4G3B,C;IyF5GnB,C;uGAE9E,yB;MAAA,iE;MAAA,wC;MAAA,mC;QAMI
,YAAy,kBAAK,KAAL,C;QACZ,IAAI,aAAJ,C;UAAmB,OAAO,KAAM,W;QAEhC,gBAAGB,C;QACHB,aAAa,K
AAM,O;QACnB,SAAS,mBAAc,MAAd,C;;UAEL,iBAaiB,oB;UACjB,EAAG,gBAAO,KAAP,EAAC,SAAd,EAAY
B,UAAW,MAAM,MAA1C,C;UACH,EAAG,gBAAO,UAAU,UAAV,CAAP,C;UACH,YAAy,UAAW,MAAM,aA
AjB,GAAgC,CAAhC,I;UACZ,QAAQ,UAAW,O;;QACd,oBAAY,MAAZ,IAAsB,aAAtB,C;QAET,IAAI,YAAy,M
AAhB,C;UACI,EAAG,gBAAO,KAAP,EAAC,SAAd,EAAYB,MAAZB,C;;QAGP,OAAO,EAAG,W;MACd,C;KAZ
BA,C;;MAiCmC,gBAAR,Y;MpGupCpB,kBAAM,iBAaA,mCAAwB,EAaxB,CAAb,C;MAqEA,Q;MAAA,2B;MA
Ab,OAAa,cAAb,C;QAAa,sB;QACT,WAAy,WAAc,IoG7tCc,MpG6tC5B,C;;MoG7tCZ,uBAAGD,epG8tC7C,WoG
9tC6C,EAAYB,EAazB,C;MACHD,OAAO,KAAM,WzFwEgF,SyFxEvD,WAAO,YAAP,EAAGB,gBAAhB,CzFwE
uD,EyFxEpB,WzFwEoB,C;IyFvEjG,C;;MAS6C,qB;QAAA,QAAa,C;MAMxC,Q;M1GvHIB,IAAI,E0GkHQ,SAAS
,C1GIHjB,CAAJ,C;QACI,c0GiHsB,8C;Q1GhHtB,MAAM,gCAAYB,OAAQ,WAAjC,C;;MIOEH,SsGuCW,qBAAQ
,KAAR,C;MAAd,cAAuC,UAAAS,CAAb,GAAgB,EAAhB,GAA2B,OAAH,EAAG,EAak,QAAQ,CAAR,IAAL,C;
MAC9D,ajH9DgD,gB;MiH+DhD,gBAAGB,C;MAEF,yB;MAAd,OAAc,cAAd,C;QAAC,uB;QACV,MAAO,WAAU
,mBAAN,KAAM,EAAY,SAAZ,EAAuB,KAAM,MAAM,MAAnC,CAA0C,WAApD,C;QACP,YAAy,KAAM,MA
AM,aAAZ,GAA2B,CAA3B,I;;MAEHb,MAAO,WAAU,mBAAN,KAAM,EAAY,SAAZ,EAAuB,KAAM,OAA7B,
CAAqC,WAA/C,C;MACP,OAAO,M;IACX,C;;MASyC,OAAA,oBAAc,W;IAAd,C;;MAEzC,+B;MAMBI,uBAA4
B,WAAO,wBAAP,EAakC,GAAIC,C;MAC5B,2BAAGC,WAAO,KAAP,EAaiB,GAAjB,C;IAPBpC,C;;MAK4D,o
BAAM,oBAAO,OAAP,CAAN,C;IAAA,C;;MAMJ,OAAA,OzF4ByC,SyF5BnB,oBzF4BmB,EyF5BJ,MzF4BI,C;Iy
F5BzC,C;;MAMW,OAAA,OzFsB8B,SyFtBR,wBzFsBQ,EyFtBW,MzFsBX,C;IyFtB9B,C;;MAjBnE,sC;QAAA,q
B;;MAAA,+B;IAAA,C;;MAjIA,+C;MACKe,kBAAK,OAAL,EAAC,MAAM,MAAN,CAAd,C;MADIE,Y;IAAA,C;
;MAGA,+C;MAC6C,kBAAK,OAAL,EAAC,UAAAd,C;MAD7C,Y;IAAA,C;;MAyJ8D,iBAAM,OAAN,EAae,MAA
M,MAAN,CAAF,C;IAAA,C;;MAKrB,iBAAM,OAAN,EAae,UAAf,C;IAAA,C;;MAWIC,kC;MAAA,kC;MAAA,k
C;MAAA,kC;MACH,uBAA+B,a;MAI/B,sF;MAOA,sBAA0C,I;IAZvC,C;;MACH,2B;IAAA,C;;MAEY,qBAAA,k
BJtK8C,CIsKxC,CJtKwC,CIsK9C,C;IAAA,C;;MAEZ,4B;IAAA,C;;MAY2B,kC;MAAS,uB;IAAT,C;;MACoB,OA
AA,kBAAM,O;IAAN,C;;MACQ,Q;MAAA,eAAA,kBJtLG,CIsLG,KJtLH,CIsLH,mBAAGB,E;IAAhB,C;;MAH/C,
IAAI,2BAAJ,C;QACI,yH;MAKJ,OAAO,kC;IACX,C;;MAEGC,OAAc,SAAd,kBAAC,EAAS,kBAAT,EAAoB,kBA
AM,UAAV,GAAqB,kBAAM,MAAN,GAAc,CAAd,IAArB,GAA0C,kBAAM,aAN,GAAqB,CAArB,IAA1D,C;IA
Ad,C;;MApBQ,kC;MAA+B,6B;IAA/B,C;;MACT,OAAA,kBAAM,O;IAAN,C;;MAC2C,qB;QAAE,yBAAK,EAAL

,C;MAAF,C;IAAA,C;;MAAzB,OAAqB,OAAb,aAAR,oBAAQ,CAAA,EAAl,iEAAJ,CAAIb,W;IAAtC,C;;MACL,Q;MAAA,eAAA,kBJ3KU,Ci2KJ,KJ3KI,Ci2KV,YAAoB,oBAApB,O;IAAA,C;;;MABpD,sBAAiB,I;MACjB,YAA Y,eAAK,KAAL,C;MACZ,IAAI,aAAJ,C;QAAMb,OAAO,I;MAC1B,YAAY,aAAA,KAAM,MAAN,EAAa,sBAAY ,CAAZ,IAAb,C;MAEZ,iE;IA2BJ,C;;MvGpOsG,4C;IAAtG,C;IAAA,+D;MAC4C,+BAAW,CAAX,EAAc,CAAd,C; IAAA,C;IAD5C,oF;;McEiB,Q;MADb,aAAa,E;MACb,wBAAa,KAAb,gB;QAAa,WAAb,UAAa,KAAb,O;QACI,8B AAU,IAAV,C;;MAEJ,OAAO,M;IACX,C;;MAa+B,Q;MAH3B,IAAI,SAAS,CAAT,IAAc,SAAS,CAAvB,IAA4B,C AAA,KAAM,OAAN,GAAa,MAAb,QAAsB,MAAtD,C;QACI,MAAM,8BAA0B,WAAAS,KAAM,OAAf,kBAA+B, MAA/B,kBAAgD,MAA1E,C;MACV,aAAa,E;MACc,gBAAS,MAAT,I;MAA3B,iBAAc,MAAd,wB;QACI,8BAAU ,MAAM,KAAN,CAAV,C;;MAEJ,OAAO,M;IACX,C;;MASiB,Q;MADb,aAAa,E;MACb,wBAAa,SAAb,gB;QAAa, WAAb,UAAa,SAAb,O;QACI,8BAAU,IAAV,C;;MAEJ,OAAO,M;IACX,C;;Mac2C,0B;QAAA,aAAkB,C;MAAG, wB;QAAA,WAAgB,SAAK,O;MACjF,oCAAA,4BAAMb,UAAAnB,EAA+B,QAA/B,EAAYC,SAAK,OAA9C,C;MA Cb,aAAa,E;MACb,iBAAc,UAAAd,UAA+B,QAA/B,U;QACI,8BAAU,UAAK,KAAL,CAAV,C;;MAEJ,OAAO,M;I ACX,C;;MAQ6B,qB;QAAE,+CAAI,EAAJ,E;MAAF,C;IAAA,C;;MAAzB,OAAO,kBAAU,gBAAV,EAakB,+BA AlB,C;IACX,C;;MAGb4C,qB;QAAE,+CAAI,qBAAa,EAAb,IAAJ,E;MAAF,C;IAAA,C;;MAFP,0B;QAAA,aAAkB ,C;MAAG,wB;QAAA,WAAgB,SAAK,O;MAC3E,oCAAA,4BAAMb,UAAAnB,EAA+B,QAA/B,EAAYC,gBAAzC, C;MACb,OAAO,kBAAU,WAAW,UAAAX,IAAV,EAAiC,2CAAjC,C;IACX,C;;MAUI,OAAO,WAAW,SAAX,EAA iB,CAAjB,EAAoB,gBAApB,EAA0B,KAA1B,C;IACX,C;;MAiBI,0B;QAAA,aAAkB,C;MACiB,wB;QAAA,WAA gB,SAAK,O;MACrB,sC;QAAA,yBAaKc,K;MAEiC,oCAAA,4BAAMb,UAAAnB,EAA+B,QAA/B,EAAYC,SAAK, OAA9C,C;MACb,OAAO,WAAW,SAAX,EAAiB,UAAjB,EAA6B,QAA7B,EAAuC,sBAAvC,C;IACX,C;;MAUI, OAAO,WAAW,SAAX,EAAiB,CAAjB,EAAoB,gBAApB,EAA4B,KAA5B,C;IACX,C;;MAiBI,0B;QAAA,aAAkB, C;MACiB,wB;QAAA,WAAgB,SAAK,O;MACrB,sC;QAAA,yBAaKc,K;MAEiC,oCAAA,4BAAMb,UAAAnB,EAA +B,QAA/B,EAAYC,gBAAzC,C;MACb,OAAO,WAAW,SAAX,EAAiB,UAAjB,EAA6B,QAA7B,EAAuC,sBAAvC ,C;IACX,C;;MAQwD,OAAA,SAAY,c;IAAZ,C;;MAQA,OAAA,SAAY,c;IAAZ,C;;MAG4B,OAAA,SAAY,SAAQ, GAAR,EAAa,SAAb,C;IAAZ,C;;MAGI,OAAA,SAAY,aAAY,GAAZ,EAAiB,SAAjB,C;IAAZ,C;;MAGP,OAAA,S AAY,YAAW,CAAX,EAAc,QAAd,C;IAAZ,C;;MAGjB,OAAA,SAAY,UAAS,CAAT,C;IAAZ,C;;MAGK,OAAA,S AAY,WAAU,UAAV,C;IAAZ,C;;MAGe,OAAA,SAAY,WAAU,UAAV,EAAsB,QAAtB,C;IAAZ,C;;MAG7B,OAA A,SAAY,QAAO,GAAP,C;IAAZ,C;;MAGS,OAAA,SAAY,OAAM,KAAN,C;IAAZ,C;;MAMyB,OAAA,SAAY,SA AQ,OAAR,EAAiB,WAAjB,C;IAAZ,C;;MAiVc,0B;QAAA,aAAsB,K;MACpE,IAAI,UAAJ,C;QACI,SAAS,SAAK, O;QACd,SAAS,KAAM,O;QACf,UZjCG,MAAK,KYiCQ,EZjCR,EYiCY,EZjCZ,C;QYkCR,IAAI,QAAO,CAAX,C ;UAAc,OAAO,KAAK,EAAL,I;QACrB,YAAY,C;QACZ,OAAO,IAAP,C;UACI,UZrCD,MAAK,KYqCY,QAAQ,E AAR,IZrCZ,EYqCwB,GZrCxB,C;UYsCJ,SAAS,SAzB2E,WaYb5D,KAZb4D,EAYBrD,GAzBqD,C;UA0BpF,SAA S,KA1B2E,WA0B3D,KA1B2D,EA0BpD,GA1BoD,C;UA2BpF,IAAI,YAAM,EAAN,CAAJ,C;YACI,KAAK,EAtd +C,c;YAuDpD,KAAK,EAvd+C,c;YAwDpD,IAAI,YAAM,EAAN,CAAJ,C;cACI,KAAK,EAjD2C,c;cAkDhD,KA AK,EAId2C,c;cAmDhD,IAAI,YAAM,EAAN,CAAJ,C;gBACI,OAAU,iBAAH,EAAG,EAAU,EAAV,C;;;UAItB,I AAI,QAAO,GAAX,C;YAAgB,K;UACHb,QAAQ,G;;QAEZ,OAAO,KAAK,EAAL,I;;;QAEp,OAAO,4BAAU,KAA V,C;;IAEF,C;;MAGyE,OAAE,UAAF,CAAE,EAAU,CAAV,EAA0B,IAA1B,C;IAAgC,C;;MAiF,oC;IAAA,C;;M0 FhPqE,O1F6Ke,kB0F7KE,oBAAH,EAAG,C1F6KF,E0F7Kc,S1F6Kd,C;I0F7Kf,C;;MAGI,O1F6Ke,sB0F7KM,oB AAH,EAAG,C1F6KN,E0F7KkB,S1F6KIB,C;I0F7Kf,C;;MAMjC,0B;QAAA,aAAsB,K;MACiE,IAAI,CAAC,UAA L,C;QACI,O1FwKqF,qB0F7K7D,M1FwK6D,E0F7KrD,C1FwKqD,C;;;Q0FtKrF,OAAO,yBAAc,CAAd,EAAiB,M AAjB,EAAyB,CAAzB,EAA4B,MAAO,OAANc,EAA2C,UAA3C,C;IACf,C;;MAMqE,0B;QAAA,aAAsB,K;MAC vF,IAAI,CAAC,UAAAL,C;QACI,O1F6JqF,qB0F7J7D,M1F6J6D,E0F7JrD,U1F6JqD,C;;;Q0F3JrF,OAAO,yBAAc,U AAd,EAA0B,MAA1B,EAAkC,CAAIc,EAAqC,MAAO,OAA5C,EAAoD,UAApD,C;IACf,C;;MAMkD,0B;QAAA, aAAsB,K;MACpE,IAAI,CAAC,UAAAL,C;QACI,O1FqJoE,mB0FrJ9C,M1FqJ8C,C;;;Q0FnJpE,OAAO,yBAAc,mB AAS,MAAO,OAAhB,IAAd,EAAc,MAAtC,EAA8C,CAA9C,EAAiD,MAAO,OAAXD,EAAgE,UAAhE,C;IACf,C ;;MAiI,aAAa,S1F0J2D,O0F1JhD,K1F0JgD,C;M0FzJxE,OAAO,kBAAkB,MAAO,OAAP,KAAe,C;IAC5C,C;;MA EoD,4BAAU,CAAV,IAA+D,QAA3C,6BAAJ,GAAoB,SAAPB,GAA8B,SAAK,WAAAY,EAAQ,eAAR,C;IAA/D,C ;;MAGH,0B;QAAA,aAAsB,K;MACnE,Q;MAAA,IAAI,iBAAJ,C;QAAA,OACI,a;;QACC,U;QAAA,IAAI,CAAC,U AAL,C;UAAA,SACD,kBAAQ,KAAR,C;;UAEA,0B;UAAA,W;YAAiB,gBAAA,S1FqH2C,c0FrH3C,EAAsB,K1F

qHqB,c0FrH3C,C;;UAHhB,e;;QAFL,a;;MAAA,W;IAAA,C;;MAS8G,0B;QAAA,aAAsB,K;MACpI,oCAAkB,UAA
IB,EAA8B,KAA9B,EAAqC,WAArC,EAaKd,MAAID,EAA0D,UAA1D,C;IAAA,C;;MAUA,O3FmKmD,mBAAS,
C2FnK5D,G1FwH4F,oB0FxFzD,C1FwHyD,E0FxFtD,C1FwHsD,CA1B5B,c0F9FvC,G1FqHoD,oB0FrHV,C1FqH
U,C0FrH7E,GAA2E,S;IAC/E,C;;MASI,O3FyJmD,mBAAS,C2FzJ5D,G1F8G4F,oB0F9GzD,C1F8GyD,E0F9GtD,
C1F8GsD,CA1B5B,c0F5FvC,G1F2GoD,oB0F3GV,C1F2GU,C0F3G7E,GAA2E,S;IAC/E,C;;MAQiB,IAAN,I;M3G
9DP,IAAI,E2G6DI,KAAK,C3G7DT,CAAJ,C;QACI,c2G4Dc,oD;Q3G3Dd,MAAM,gCAAyB,OAAQ,WAAJ,C,C;;
M2G4DH,QAAM,CAAN,C;aACH,C;UAAK,S;UAAL,K;aACA,C;UAAU,OAAL,SAAK,W;UAAV,K;gBAEI,aAA
a,E;UACb,IAAI,E3FmIoC,qBAAU,C2FnI9C,CAAJ,C;YACI,QAAQ,SAAK,W;YACb,YAAY,C;YACZ,OAAO,IA
AP,C;cACI,IAAI,CAAC,QAAU,CAAX,MAAiB,CAArB,C;gBACI,UAAU,C;;cAEd,QAAQ,UAAW,C;cACnB,IA
AI,UAAAS,CAAb,C;gBACI,K;;cAEJ,KAAK,C;;;UAGb,OAAO,M;;MAAnBf,W;IASBJ,C;;MAGqE,0B;QAAA,aAAs
B,K;MACvF,O1FuFiG,kB0FvFnF,WAAO,6BAAM,gBAAO,QAAP,CAAb,EAAMC,UAAJ,GAAgB,IAAhB,GAA0
B,GAAzD,C1FuFmF,E0FvFpB,6BAAM,2BAaKB,QAAIB,C1FuFc,C;I0FvFjG,C;;MAG2D,0B;QAAA,aAAsB,K;
MACjF,O1FmFiG,kB0FnFnF,WAAO,6BAAM,gBAaE,oBAAR,OAAQ,CAAf,CAAb,EAA6C,UAAJ,GAAgB,IAA
hB,GAA0B,GAAAnE,C1FmFmF,E0FnFF,oBAAR,OAAQ,C1FmFE,C;I0FnFjG,C;;MAGsE,0B;QAAA,aAAsB,K;M
AC5F,O1F+EIg,kB0F/EnF,WAAO,6BAAM,gBAAO,QAAP,CAAb,EAAMC,UAAJ,GAAgB,GAAhB,GAAyB,EA
AxD,C1F+EmF,E0F/EtB,6BAAM,2BAaKB,QAAIB,C1F+EgB,C;I0F/EjG,C;;MAGgE,0B;QAAA,aAAsB,K;MACt
F,O1F2EIg,kB0F3EnF,WAAO,6BAAM,gBAaE,oBAAR,OAAQ,CAAf,CAAb,EAA6C,UAAJ,GAAgB,GAAhB,G
AAyB,EAAIE,C1F2EmF,E0F3EJ,oBAAR,OAAQ,C1F2EI,C;I0F3EjG,C;;;MCzH8B,uB;QAAA,UAAkB,E;MAOh
D,gBAA6B,O;IAPA,C;;MAUjB,OAAO,aAAY,O;IAAnB,C;;MAGD,gBAAP,a;M7FwEG,Q;MAAA,I6FxEc,K7Fw
EV,IAAS,CAAT,I6FxEU,K7FwEI,IAAS,2BAA3B,C;QAAA,OAAc,qB6FxEb,K7FwEwB,C;;Q6FxEf,MAAM,8
BAA0B,mCAAyB,WAAzB,MAA1B,C;;MAAhC,W;IAAA,C;;MAE4E,OAAA,a3FyKY,W2FzKK,U3FyKL,E2FzK
iB,Q3FyKjB,C;I2FzKZ,C;;MAG5E,qCAAU,CAAV,C;MACA,OAAO,I;IACX,C;;MAGI,iBAAc,SAAJ,GAAL,C;M
ACd,OAAO,I;IACX,C;;MAGI,iBAAc,SAAJ,GAAL,C3F4J0E,W2F5JrD,K3F4JqD,E2F5J9C,G3F4J8C,C;M2F3JxF,
OAAO,I;IACX,C;;MAGI,iBAAc,SAAJ,GAAL,C;MACd,OAAO,I;IACX,C;;MAGI,gBAAGB,aAAY,OAAM,EAAN
,CAAU,UAAU,MAAK,EAAL,C;MACHd,OAAO,I;IACX,C;;MASI,gBAAS,E;MACT,OAAO,I;IACX,C;;MAEkC,
oB;IAAA,C;;MApDIC,uD;MAAoC,2B;MAApC,Y;IAAA,C;;MAEA,uD;MAA4C,0BAAK,OAAQ,WAAb,C;MAA
5C,Y;IAAA,C;;MAEA,uD;MAAuB,0BAAK,EAAL,C;MAAvB,Y;IAAA,C;;MA4D4D,OAAA,SAAK,Q;IAAL,C;;u
BAQgB,O;;IAA/B,C;;MAC7C,kE;MAAuB,qCAAK,IAAL,C;MAAvB,Y;IAAA,C;;MC9EA,IAAI,gBAAJ,C;QAAs
B,MAAM,6BAAYB,qCAAKC,QAAQ,CAAR,IAAIC,CAAZB,C;MAC5B,OAAO,CAAC,IAAD,I;IACX,C;;MAUI,I
AAI,EAAS,KAAT,oBAAiB,KAAjB,KAA2B,SAAS,QAAxC,C;QACI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAo
B,gBAAPB,C;;MAEX,UAAU,kBAAO,KAAP,CAAc,I;MACxB,IAAI,EAAQ,KAAR,kBAAGB,KAAhB,CAAJ,C;Q
ACI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAAPB,C;;MAEX,OAAO,SAAW,CAAC,OAAS,IAAV,KAAq
B,EAAhC,IAAwC,MAAQ,I;IAC3D,C;;MAUI,IAAI,SAAU,EA AV,MAAkB,CAAIB,IAAuB,SAAS,QAApC,C;QA
CI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAAPB,C;;MAEX,YAAY,KAAa,CAAP,KAAO,C;MACzB,IAA
I,SAAU,GA AV,MAAkB,GAAtB,C;QACI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAAPB,C;;MAEX,OAA
Q,SAAU,CAAX,GAakB,KAAIB,GAA4B,I;IACvC,C;;MAWI,IAAI,SAAS,QAAb,C;QACI,OAAO,UAAU,CAAV,
EAAa,KAAb,EAAoB,gBAAPB,C;;MAGX,YAAY,KAAa,CAAP,KAAO,C;MACzB,IAAI,SAAU,EA AV,MAAiB,
CAArB,C;QACI,IAAI,SAAU,GA AV,MAAkB,GAAtB,C;UAEI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBA
APB,C;;YAER,IAAI,SAAU,EA AV,MAAiB,EAAR,B,C;QACH,IAAI,SAAU,GA AV,MAAkB,GAAtB,C;UAEI,OA
AO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAAPB,C;;YAER,IAAI,SAAU,GA AV,MAAkB,GAAtB,C;QACH,OA
AO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAAPB,C;;MAGX,IAAI,SAAQ,CAAR,UAAa,QAAjB,C;QACI,OAAO
,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAAPB,C;;MAEX,YAAY,KAAiB,CAAX,QAAQ,CAAR,IAAW,C;MAC7
B,IAAI,SAAU,GA AV,MAAkB,GAAtB,C;QACI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAAPB,C;;MAG
X,OAAQ,SAAU,EAAX,GAAoB,SAAU,CAA9B,GAAqC,KAArC,GAA+C,O;IAC1D,C;;MAWI,IAAI,SAAS,QAA
b,C;QACI,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAAPB,C;;MAGJ,YAAY,KAAa,CAAP,KAAO,C;MACzB,IAAI
,SAAU,EA AV,MAAiB,CAArB,C;QACI,IAAI,SAAU,GA AV,KAakB,GAAtB,C;UAEI,OAAO,UAAU,CAAV,EA
Aa,KAAb,EAAoB,gBAAPB,C;;YAER,IAAI,SAAU,EA AV,MAAiB,CAArB,C;QACH,IAAI,SAAU,GA AV,MAA
kB,GAAtB,C;UAEI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAAPB,C;;YAER,IAAI,SAAU,EA AV,IAAgB

,CAApB,C;QACH,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;YACJ,IAAI,SAAU,GAAV,MAAkB,GAAtB,C;QACH,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAGX,IAAI,SAAQ,CAAR,UAAa,QAAjB,C;QACI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAEX,YAAY,KAAiB,CAAX,QAAQ,CAAR,IAAW,C;MAC7B,IAAI,SAAU,GAAV,MAAkB,GAAtB,C;QACI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAGX,IAAI,SAAQ,CAAR,UAAa,QAAjB,C;QACI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAEX,YAAY,KAAiB,CAAX,QAAQ,CAAR,IAAW,C;MAC7B,IAAI,SAAU,GAAV,MAAkB,GAAtB,C;QACI,OAAO,UAAU,CAAV,EAAa,KAAb,EAAoB,gBAApB,C;;MAEX,OAAQ,SAAU,EAAX,GAAoB,SAAU,EAA9B,GAAuC,SAAU,CAAjD,GAAwD,KAAxD,GAakE,O;IAC7E,C;;;MAoC0B,UAGJ,MAHI,EAKJ,MALI,EAMJ,MANI,EASJ,MATI,EAUJ,MAVI,EAWJ,MAXI,EAgBA,MAhBA,EAiBA,MAjBA,EAkBA,MAiBA,EAoBA,MApBA,EAqBA,OArBA,EAsBA,OAtBA,EAuBA,O;M7G/JtB,IAAI,E6GiI,cAAc,CAAd,IAAmB,YAAY,MAAO,OAAtC,IAAgD,cAAc,Q7GjIIE,CAAJ,C;QACI,cAda,qB;QAeb,MAAM,gCAAYB,OAAQ,WAAjC,C;;M6GiIV,YAAY,cAAU,CAAC,WAAW,UAAW,IAAD,IAA0B,CAA1B,IAAV,C;MACZ,gBAAgB,C;MACHb,gBAAgB,U;MAEhB,OAAO,YAAY,QAAnB,C;QACI,WAAW,mBAAO,gBAAP,EAAO,wBAAP,QAAoB,I;QAE3B,WAAO,GAAP,C;UACI,MAAM,kBAAN,EAAM,0BAAN,YAA0B,OAAL,IAAK,C;;cAC9B,WAAO,IAAP,C;UACI,MAAM,kBAAN,EAAM,0BAAN,YAA4C,OAArB,QAAS,CAAV,GAAGB,GAAM,C;UAC5C,MAAM,kBAAN,EAAM,0BAAN,YAA+C,OAAxB,OAAS,EAAV,GAAMB,GAAM,C;;cAEnD,WAAO,KAAP,IAAiB,QAAQ,KAAzB,C;UACI,MAAM,kBAAN,EAAM,0BAAN,YAA6C,OAAtB,QAAS,EAAV,GAAiB,GAAM,C;UAC7C,MAAM,kBAAN,EAAM,0BAAN,YAAuD,OAA/B,QAAS,CAAV,GAAiB,EAAlB,GAA2B,GAAM,C;UACvD,MAAM,kBAAN,EAAM,0BAAN,YAA+C,OAAxB,OAAS,EAAV,GAAMB,GAAM,C;;UAG/C,gBAAgB,uBAAuB,MAAvB,EAA+B,IAA/B,EAaqC,SAArC,EAAGD,QAAdD,EAA0D,gBAA1D,C;UACbB,IAAI,aAAa,CAAjB,C;YACI,MAAM,kBAAN,EAAM,0BAAN,YAAqB,0BAA0B,CAA1B,C;YACrB,MAAM,kBAAN,EAAM,0BAAN,YAAqB,0BAA0B,CAA1B,C;;YAErB,MAAM,kBAAN,EAAM,0BAAN,YAAkD,OAA3B,aAAc,EAaf,GAAsB,GAAM,C;YACID,MAAM,mBAAN,EAAM,2BAAN,aAA6D,OAArC,aAAc,EAaf,GAAuB,EAAXB,GAaiC,GAAM,C;YAC7D,MAAM,mBAAN,EAAM,2BAAN,aAA4D,OAApC,aAAc,CAAf,GAAsB,EAavB,GAAGC,GAAM,C;YAC5D,MAAM,mBAAN,EAAM,2BAAN,aAAoD,OAA7B,YAAc,EAaf,GAAwB,GAAM,C;YACpD,6B;;;MAMhB,OAAW,KAAM,OAAN,KAAc,SAAIB,GAA6B,KAA7B,GAA8C,UAAAN,KAAM,EAAO,SAAP,C;IACzD,C;;;MAwByB,Q;M7G/LrB,IAAI,E6GyLI,cAAc,CAAd,IAAmB,YAAY,KAAM,OAArC,IAA6C,cAAc,Q7GzL/D,CAAJ,C;QACI,cAda,qB;QAeb,MAAM,gCAAYB,OAAQ,WAAjC,C;;M6GyLV,gBAAgB,U;MACHb,oBAAoB,sB;MAEpB,OAAO,YAAY,QAAnB,C;QACI,WAAW,KAAmB,CAAb,gBAAa,EAAb,wBAAa,O;QAE1B,YAAQ,CAAR,C;UACI,aAAc,gBAAAY,OAAL,IAAK,CAAZ,C;aACIB,YAAS,CAAT,KAAc,EAAd,C;UACI,WAAW,eAAe,KAAf,EAAsB,IAAtB,EAA4B,SAA5B,EAauC,QAAvC,EAaiD,gBAAjD,C;UACX,IAAI,QAAQ,CAAZ,C;YACI,aAAc,gBAAO,gBAAP,C;YACd,yBAAa,CAAC,IAAD,IAAb,K;;YAEA,aAAc,gBAAAY,OAAL,IAAK,CAAZ,C;YACd,wBAAa,CAAb,I;;cAGR,YAAS,CAAT,KAAc,EAAd,C;UACI,aAAW,eAAe,KAAf,EAAsB,IAAtB,EAA4B,SAA5B,EAauC,QAAvC,EAaiD,gBAAjD,C;UACX,IAAI,UAAQ,CAAZ,C;YACI,aAAc,gBAAO,gBAAP,C;YACd,yBAAa,CAAC,MAAD,IAAb,K;;YAEA,aAAc,gBAAAY,OAAL,MAAK,CAAZ,C;YACd,wBAAa,CAAb,I;;cAGR,YAAS,CAAT,KAAc,EAAd,C;UACI,aAAW,eAAe,KAAf,EAAsB,IAAtB,EAA4B,SAA5B,EAauC,QAAvC,EAaiD,gBAAjD,C;UACX,IAAI,UAAQ,CAAZ,C;YACI,aAAc,gBAAO,gBAAP,C;YACd,yBAAa,CAAC,MAAD,IAAb,K;;YAEA,WAAW,MAAD,GAAQ,KAAR,IAAqB,EAARb,GAA2B,K;YACtC,UAAW,SAAS,IAAV,GAAoB,K;YAC9B,aAAc,gBAAAY,OAAL,IAAK,CAAZ,C;YACd,aAAc,gBAAW,OAAl,GAAl,C;YACd,wBAAa,CAAb,I;;UAIJ,UAAU,CAAV,EAAa,SAAb,EAawB,gBAAxB,C;UACA,aAAc,gBAAO,gBAAP,C;;MAK1B,OAAO,aAAc,W;IACzB,C;;MCxQA,e;MAEsC,0B;MAFtC,iB;MAAA,uB;IAEqC,C;;MAFrC,sC;MAAA,C;MAGI,uEAGY,GAHZ,C;MAIA,yEAGa,MAHb,C;MAIA,yEAGa,SAHb,C;MAIA,+DAGQ,KAHR,C;MAIA,+DAGQ,MAHR,C;MAIA,2DAGM,MAHN,C;MAIA,yDAGK,OAHL,C;IA3BJ,C;;MAGI,yB;MAAA,wC;IAAA,C;;MAIA,yB;MAAA,yC;IAAA,C;;MAIA,yB;MAAA,yC;IAAA,C;;MAIA,yB;MAAA,oC;IAAA,C;;MAIA,yB;MAAA,oC;IAAA,C;;MAIA,yB;MAAA,kC;IAAA,C;;MAIA,yB;MAAA,iC;IAAA,C;;MA3BJ,4Q;IAAA,C;;MAAA,a;aAAA,a;UAAA,6C;aAAA,c;UAAA,8C;aAAA,c;UAAA,8C;aAAA,S;UAAA,yC;aAAA,S;UAAA,yC;aAAA,O;UAAA,uC;aAAA,M;UAAA,sC;gBAAA,6D;;IAAA,C;;MAqCW,Q;MADP,0BAA2C,iBAAjB,UAAW,cAAM,EAau,UAAW,cAARb,C;MAEvC,0BAAsB,CAAtB,C;QAA2B,gBAAS,UAAW,cAAX,GAAMB,UAAW,cAAV

C,C;WAC3B,0BAAsB,CAAtB,C;QAA2B,gBAAS,UAAW,cAAX,GAAMB,UAAW,cAAvC,C;;QACnB,Y;MAHZ,W;IAKJ,C;;MCtCkB,Q;MAAA,IAAI,aAAY,CAAhB,C;QACV,Y;;;QAEA,UjByY8C,MAAW,KiBzY/C,IjByY+C,EiBzYtC,QjByYsC,C;QiBvYzD,OAAe,WjB6U2B,MAAW,KiB7U5B,KjB6U4B,CiB7UhC,GAAa,GAAnB,CAAF,GAAYC,GAAzC,GjB0V2C,MAAW,MiB1VF,KjB0VE,C;;MiB/V1D,kB;MAOA,OAAe,OAAAY,SAAQ,QAAR,C;IAC/B,C;;MAGI,OAAa,KAAY,gBA Ae,OAAf,EAAwB,MAAK,4BAA2B,QAA3B,CAAL,EAAxB,C;IAC7B,C;;MAGI,OAAa,KAAY,eAAc,CAAd,C;IAC7B,C;;MCjBA,yB;MAUuB,wB;MALf,aAAR,OAAO,OAAQ,KAAL,WAAAY,IAAG,OAAO,SAAX,IAAwB,CAAC,CAAC,OAAO,SAAS,K;MADpE,qBAGQ,MAHR,GAIQ,gBAAAY,OAAZ,CAJR,GAMQ,qBAAW,OAAAX,IAAA,+CnGnBZ,4BmGmBY,4BACO,0B;IAXnB,C;;MAewC,OAAA,kBAAY,U;IAAZ,C;;;MAfxC,gC;QAAA,e;;MAAA,yB;IAAA,C;;MAwB2B,sB;IAAD,C;;MAEc,wC;MAAS,oB;MACzC,iBAAGB,wBAAQ,S;IADQ,C;;M5G2E7B,Q4GxEC,6BAAQ,QAAO,cAAP,C;MAAYB,c7HfIC,EAAI,CAAJ,C;M6He2C,Y7HyD3C,EAAI,CAAJ,C;M6HzDC,OAA4D,cAAR,OAAQ,CAAR,aAAwB,kBAAN,KAAM,CAAxB,C;IAApD,C;;MAH4B,kD;IAAA,C;;MAMF,gC;IAAA,C;;MAK0B,uE;MAAhC,8B;IAAD,C;;MACG,OAAA,gBAAAY,M;IAAZ,C;;MACI,sC;IAAA,C;;MAGtC,4B;MAE+B,uE;IAF/B,C;;MAGkC,OAAe,U;IAAf,C;;MACI,0B;IAAA,C;;;MAJtC,mC;QAAA,kB;;MAAA,4B;IAAA,C;;MCKyC,qB;QAAA,QAaKB,I;MAAM,qB;QAAA,QAaKB,I;MAAM,uB;QAAA,UAAoB,K;MAAO,yB;QAAA,YAAsB,I;MAAM,kC;QAAA,qBAA+B,I;MAAM,qC;QAAA,wBAaK,C;MAAO,+C;QAAA,kCAA4C,K;MAAO,4C;QAAA,+BAAYC,K;MACtT,QAAQ,E;MAER,EAAE,OAAf,IAAa,K;MACb,EAAE,OAAf,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,WAAf,IAAiB,S;MACjB,EAAE,oBAAF,IAA0B,kB;MACiB,EAAE,uBAAF,IAA6B,qB;MAC7B,EAAE,iCAAF,IAAuC,+B;MACvC,EAAE,8BAAF,IAAoC,4B;MAEpC,OAAO,C;IACX,C;;MAy0BwC,6B;QAAA,gBAAyB,E;MAAI,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/I,QAAQ,E;MAER,EAAE,eAAF,IAAqB,a;MACrB,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACiB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MA2CkF,iBAAY,KAAZ,C;IAAA,C;;MACa,UAAAY,KAAZ,IAAqB,K;IAAO,C;;MASBxC,iBAAY,KAAZ,C;IAAA,C;;MACa,UAAAY,KAAZ,IAAqB,K;IAAO,C;;MASB/C,iBAAY,KAAZ,C;IAAA,C;;MACa,UAAAY,KAAZ,IAAqB,K;IAAO,C;;MASBxC,iBAAY,KAAZ,C;IAAA,C;;MACa,UAAAY,KAAZ,IAAqB,K;IAAO,C;;MASB5C,iBAAY,KAAZ,C;IAAA,C;;MACa,UAAAY,KAAZ,IAAqB,K;IAAO,C;;MASBxC,iBAAY,KAAZ,C;IAAA,C;;MACa,UAAAY,KAAZ,IAAqB,K;IAAO,C;;MASBtC,iBAAY,KAAZ,C;IAAA,C;;MACa,UAAAY,KAAZ,IAAqB,K;IAAO,C;;MASBxC,iBAAY,KAAZ,C;IAAA,C;;MACa,UAAAY,KAAZ,IAAqB,K;IAAO,C;;MCxlc3F,6B;QAAA,gBAA+B,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACpJ,QAAQ,E;MAER,EAAE,eAAF,IAAqB,a;MACrB,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACiB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MA0BgD,mC;QAAA,sBAAgC,K;MAC5E,QAAQ,E;MAER,EAAE,qBAAF,IAA2B,mB;MAE3B,OAAO,C;IACX,C;;MCKm+F,iBAAY,KAAZ,C;IAAA,C;;MASV,iBAAY,KAAZ,C;IAAA,C;;MAGCS,iBAAY,KAAZ,C;IAAA,C;;MAGBN,iBAAY,KAAZ,C;IAAA,C;;MCzr1D,oB;QAAA,OAAgB,I;MAAM,sB;QAAA,SA Ae,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACChJ,QAAQ,E;MAER,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACiB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAGBiC,6B;QAAA,gBAA8B,I;MAAM,oB;QAAA,OAAgB,I;MAAM,sB;QAAA,SA Ae,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACvL,QAAQ,E;MAER,EAAE,eAAF,IAAqB,a;MACrB,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACiB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAkDiC,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,sB;QAAA,SAAiB,C;MAAG,uB;QAAA,UAAkB,C;MAAG,6B;QAAA,gBAA8B,I;MAAM,uB;QAAA,UAAoB,K;MAAO,wB;QAAA,WAAqB,K;MAAO,sB;QAAA,SAAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,0B;QAAA,aAAuB,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,+B;QAAA,kBAA4B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,oB;QAAA,OAAgB,I;MAAM,sB;QAAA,SA Ae,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACnVb,QAAQ,E;MAER,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,eAAF,IAAqB,a;MACr

B,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MACHb,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,eAAF,IAAqB,a;MACrB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,eAAF,IAAqB,a;MACrB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAGDoC,uB;QAAA,UAAoB,K;MAAO,wB;QAAA,WAAqB,K;MAAO,sB;QAAA,SAAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,0B;QAAA,aAAuB,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,+B;QAAA,kBAA4B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,oB;QAAA,OAAGB,I;MAAM,sB;QAAA,SA Ae,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC7IB,QAA Q,E;MAER,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MACHb,EAAE,QAAF,IAAc,M;MACd,EAAE, SAAF,IAAe,O;MACf,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,YAAF,IAA kB,U;MACIB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,eAAF,IAAqB,a;MACrB,EAAE,iBAAF,IAAuB,e;MACvB, EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,eAAF,IAAqB,a;MACrB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,oBA AAF,IAA0B,kB;MAC1B,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf ,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAkCiC,sB;QAAA,SAAk B,G;MAAK,sB;QAAA,SAAkB,G;MAAK,sB;QAAA,SAAkB,G;MAAK,yB;QAAA,YAAkB,C;MAAG,uB;QAAA, UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,sB; QAAA,SAAiB,C;MAAG,uB;QAAA,UAAkB,C;MAAG,6B;QAAA,gBAA8B,I;MAAM,uB;QAAA,UAAoB,K;MA AO,wB;QAAA,WAAqB,K;MAAO,sB;QAAA,SAAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,gC;QAAA,mBA A6B,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,0B;QAAA,aAAuB,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,6B;Q AAA,gBAA0B,K;MAAO,+B;QAAA,kBAA4B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,6B;QAAA,gBAA0B,K; MAAO,8B;QAAA,iBAA2B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,oB;QAAA,OAAGB,I;MAAM,sB;QAAA,S AAe,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC70B,QA AQ,E;MAER,EAAE,QAAF,IAAc,M;MACd,EAAE,QAAF,IAAc,M;MACd,EAAE,QAAF,IAAc,M;MACd,EAAE, WAAF,IAAiB,S;MACjB,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf, EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,eAAF,IAAqB,a; MACrB,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MACHb,EAAE,QAAF,IAAc,M;MACd,EAAE,SA AAF,IAAe,O;MACf,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,YAAF,IAAkB, U;MACIB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,eAAF,IAAqB,a;MACrB,EAAE,iBAAF,IAAuB,e;MACvB,EA AE,oBAAF,IAA0B,kB;MAC1B,EAAE,eAAF,IAAqB,a;MACrB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,oBAAF,I AA0B,kB;MAC1B,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EA AE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAoBiC,oB;QAAA,OAAGB,E ;MAAI,2B;QAAA,cAAwB,K;MAAO,oB;QAAA,OAAGB,I;MAAM,sB;QAAA,SA Ae,C;MAAG,uB;QAAA,UAAo B,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACtM,QAAQ,E;MAER,EAAE,MAAF,IAAY, I;MACZ,EAAE,aAAF,IAAmB,W;MACnB,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,S AAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MA+ C oC,mB;QAAA,MAAe,E;MAAI,oB;QAAA,OAAGB,E;MAAI,wB;QAAA,WAAiB,C;MAAG,sB;QAAA,SAAmB,K ;MAAO,2B;QAAA,cAAwB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,wB;QAAA,WAAqB,K;MAAO,sB;QAAA,S AAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,gC;QAAA,mBAA6B,K;MAAO, 0B;QAAA,aAAuB,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,+B;QAAA,kBAA4B, K;MAAO,kC;QAAA,qBAA+B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,kC;QAA A,qBAA+B,K;MAAO,oB;QAAA,OAAGB,I;MAAM,sB;QAAA,SA Ae,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B ;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACjtB,QAAQ,E;MAER,EAAE,KA AF,IAAW,G;MACX,EAA E,MAAF,IAAY,I;MACZ,EAAE,UAAF,IAAgB,Q;MACHb,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,IAAmB, W;MACnB,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MACHb,EAAE,QAAF,IAAc,M;MACd,EAAE,S AAF,IAAe,O;MACf,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,YAAF,IAAk

B,U;MACIB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,eAAF,IAAqB,a;MACrB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,eAAF,IAAqB,a;MACrB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAGBuC,oB;QAAA,OAAgB,E;MAAI,oB;QAAA,OAAgB,I;MAAM,sB;QAAA,SAAe,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC7K,QAAQ,E;MAER,EAAE,MAAF,IAAY,I;MACZ,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MC5NsF,iBAAy,IAAZ,C;IAAA,C;;MAuEF,iBAAy,IAAZ,C;IAAA,C;;MAkCsC,UAAy,KAAZ,IAAqB,M;IAAQ,C;;MA0C7D,iBAAy,IAAZ,C;IAAA,C;;MACY,UAAy,IAAZ,IAAoB,K;IAAO,C;;MAmqBnC,iBAAy,KAAZ,C;IAAA,C;;MAwBA,iBAAy,KAAZ,C;IAAA,C;;MAoBF,iBAAy,KAAZ,C;IAAA,C;;MAuBM,iBAAy,KAAZ,C;IAAA,C;;MAcJqE,qB;QAAA,QAAiD,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACIK,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAGJ4F,iBAAy,KAAZ,C;IAAA,C;;MACsB,iBAAy,IAAZ,C;IAAA,C;;MAGIpB,iBAAy,KAAZ,C;IAAA,C;;MACwB,UAAy,KAAZ,IAAqB,M;IAAQ,C;;MAstH,6B;QAAA,gBAA8B,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACjJ,QAAQ,E;MAER,EAAE,eAAF,IAAqB,a;MACrB,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAMDuC,uB;QAAA,UAAoB,K;MACvD,QAAQ,E;MAER,EAAE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;;MAoBmD,qB;QAAA,QAAkB,I;MACjE,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MAEb,OAAO,C;IACX,C;;MAstMmC,oB;QAAA,OAAgB,I;MAAM,wB;QAAA,WaorE4B,S;;MAprEwB,kB;QAAA,KAAc,E;MAAI,wB;QAAA,WAAoB,I;MAAM,sB;QAAA,SAAkB,S;MAAW,uB;QAAA,UAAoB,I;MAAM,qB;QAAA,QAAiB,I;MAAM,oB;QAAA,OAAgB,I;MACnP,QAAQ,E;MAER,EAAE,MAAF,IAAY,I;MACZ,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,IAAF,IAAU,E;MACV,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,OAAF,IAAa,K;MACb,EAAE,MAAF,IAAY,I;MAEZ,OAAO,C;IACX,C;;MAuDsD,qB;QAAA,QAAkB,I;MACpE,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MAEb,OAAO,C;IACX,C;;MAkB2C,yB;QAAA,YAAmB,S;MACID,QAAQ,E;MAER,EAAE,SAAF,IAAe,S;MAEf,OAAO,C;IACX,C;;MAGC0G,iBAAy,KAAZ,C;IAAA,C;;MA0B1E,4B;QAAA,eAA8B,I;MAAM,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,sB;QAAA,SAAiB,C;MAAG,uB;QAAA,UAAkB,C;MAAG,6B;QAAA,gBAA8B,I;MAAM,uB;QAAA,UAAoB,K;MAAO,wB;QAAA,WAAqB,K;MAAO,sB;QAAA,SAAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,0B;QAAA,aAAuB,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,+B;QAAA,kBAA4B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,oB;QAAA,OAAgB,I;MAAM,sB;QAAA,SAAe,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACtXB,QAAQ,E;MAER,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,eAAF,IAAqB,a;MACrB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,eAAF,IAAqB,a;MACrB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAqDoC,qB;QAAA,QAAc,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACII,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAoBsC,sB;QAAA,SAAkB,E;MAAI,sB;QAAA,SAAkB,E;MAAI,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC5J,QAAQ,E;MAER,EAAE,QAAF,IAAc,M;MACd,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAGB0C,yB;QAAA,YAAsB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,

WAAqB,K;MACjJ,QAAQ,E;MAER,EAAE,WAAF,IAAiB,S;MACjB,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MASeiC,uB;QAAA,UAAmB,E;MAAI,wB;QAAA,WAAoB,E;MAAI,sB;QAAA,SAAe,C;MAAG,qB;QAAA,QAAc,C;MAAG,qB;QAAA,QAAc,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACjN,QAAQ,E;MAER,EA AE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,QAAF,IAAc,M;MACd,EAAE,OAAF,IAAa,K; MACb,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF ,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAoBqE,sB;QAAA,SAAe,S;MAAW,uB;QAAA,UAAoB,K;MAAO,0B;Q AAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACzK,QAAQ,E;MAER,EAAE,SAAF,IAAe,O;MACf,EAAE,Q AAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MA EhB,OAAO,C;IACX,C;;MAmLuF,iBAAY,KAAZ,C;IAAA,C;;MACE,iBAAY,IAAZ,C;IAAA,C;;MAUE,iBAAY, KAAZ,C;IAAA,C;;MACE,iBAAY,IAAZ,C;IAAA,C;;MAaT,iBAAY,KAAZ,C;IAAA,C;;MACE,iBAAY,IAAZ,C;I AAA,C;;MA2CjD,gC;QAAA,mBA6lD8C,M;;MA7lDe,gC;QAAA,mBA5mDT,S;;MAtmDyE,oC;QAAA,uBA6mDj E,S;;MA7mD6l,2B;QAAA,cAAoB,S;MAAW,4B;QAAA,eAAqB,S;MAAW,6B;QAAA,gBAonDlO,K;;MAnnDvE, QAAQ,E;MAER,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,sBAAF,IAA4B, oB;MAC5B,EAAE,aAAF,IAAmB,W;MACnB,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,eAAF,IAAqB,a;MAErB,O AAO,C;IACX,C;;MAiCmC,oB;QAAA,OAAa,I;MAAM,sB;QAAA,SAAkB,E;MAAI,2B;QAAA,cAAuB,E;MAAI, sB;QAAA,SAAoC,I;MAAM,qB;QAAA,QAA6B,E;MAAW,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K; MAAO,wB;QAAA,WAAqB,K;MACnQ,QAAQ,E;MAER,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;M ACd,EAAE,aAAF,IAAmB,W;MACnB,EAAE,QAAF,IAAc,M;MACd,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF ,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MA4BkC,+ B;QAAA,kBAA4B,K;MAClD,QAAQ,E;MAER,EAAE,iBAAF,IAAuB,e;MAEvB,OAAO,C;IACX,C;;MAoDiC,w B;QAAA,WAAqB,K;MAAO,oB;QAAA,OAAe,C;MAAG,sB;QAAA,SAAkB,E;MAAI,uB;QAAA,UAAoB,K;MA AO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/K,QAAQ,E;MAER,EAAE,UAAF,IAAgB,Q;MAC hB,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAk B,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MA4FgC,oB;QAAA,OA64C6C,S;;MA74CL,2B ;QAAA,cCvpFe,M;;MDwpFnF,QAAQ,E;MAER,EAAE,MAAF,IAAY,I;MACZ,EAAE,aAAF,IAAmB,W;MAEnB, OAAO,C;IACX,C;;MAkDoF,iBAAY,GAZ,C;IAAA,C;;MACY,UAY,GAZ,IAAmB,K;IAAO,C;;MA8CvF,m B;QAAA,MAAe,I;MAAM,wB;QAAA,WAAoB,I;MAAM,wB;QAAA,WAAoB,I;MAAM,mB;QAAA,MAAe,E;M AAI,2B;QAAA,cAAwB,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAq B,K;MACvO,QAAQ,E;MAER,EAAE,KAAF,IAAW,G;MACX,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,UAAF,IA AgB,Q;MAChB,EAAE,KAAF,IAAW,G;MACX,EAAE,aAAF,IAAmB,W;MACnB,EAAE,SAAF,IAAe,O;MACf,E AAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MA0F4B,uB;QAAA,UAAoB ,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACtG,QAAQ,E;MAER,EAAE,SAAF,IAAe,O; MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAiBkC,sB;QAAA, SAAe,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACjL,QAA Q,E;MAER,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE, UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MASuC,uB;QAAA,UAAoB,K;MACvD,QAAQ,E;MAER,EAAE,SA AAF,IAAe,O;MAEf,OAAO,C;IACX,C;;MAY0C,uB;QAAA,UAAoB,K;MAAO,oB;QAAA,OAAiB,K;MAAO,uB;Q AAA,UAAoB,K;MAC7G,QAAQ,E;MAER,EAAE,SAAF,IAAe,O;MACf,EAAE,MAAF,IAAY,I;MACZ,EAAE,SA AAF,IAAe,O;MAEf,OAAO,C;IACX,C;;MA2DkF,iBAAY,KAAZ,C;IAAA,C;;MAUS,iBAAY,KAAZ,C;IAAA,C;;M ACE,iBAAY,IAAZ,C;IAAA,C;;MAuCd,yB;QAAA,YAAsB,K;MAAO,0B;QAAA,aAAuB,S;MAAW,6B;QAAA, gBAA0B,S;MAAW,uB;QAAA,UAAoB,K;MAAO,iC;QAAA,oBAA8B,S;MAAW,qC;QAAA,wBAAkC,S;MAAW ,+B;QAAA,kBAAkC,S;MAClR,QAAQ,E;MAER,EAAE,WAAF,IAAiB,S;MACjB,EAAE,YAAF,IAAkB,U;MACl B,EAAE,eAAF,IAAqB,a;MACrB,EAAE,SAAF,IAAe,O;MACf,EAAE,mBAAF,IAAY,iB;MACzB,EAAE,uBAAF ,IAA6B,qB;MAC7B,EAAE,iBAAF,IAAuB,e;MAEvB,OAAO,C;IACX,C;;MAgFqC,wB;QAAA,WAAqB,K;MACt D,QAAQ,E;MAER,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAwByC,mB;QAAA,MAAe,S;MACpD, QAAQ,E;MAER,EAAE,IAAF,IAAU,G;MAEV,OAAO,C;IACX,C;;MA2HI,QAAQ,E;MAER,EAAE,MAAF,IAAY ,I;MAEZ,OAAO,C;IACX,C;;MAesF,iBAAY,KAAZ,C;IAAA,C;;MACW,iBAAY,aAAZ,C;IAAA,C;;MAiMT,iBA

AY,KAAZ,C;IAAA,C;;MA4CzD,iB;QAAA,IAAa,G;MAAK,iB;QAAA,IAAa,G;MAAK,iB;QAAA,IAAa,G;MAA
K,iB;QAAA,IAAa,G;MAC9F,QAAQ,E;MAER,EAAE,GAFF,IAAS,C;MACT,EAAE,GAFF,IAAS,C;MACT,EAA
E,GAFF,IAAS,C;MACT,EAAE,GAFF,IAAS,C;MAET,OAAO,C;IACX,C;;MA0C8B,iB;QAAA,IAAa,G;MAAK,i
B;QAAA,IAAa,G;MAAK,qB;QAAA,QAAiB,G;MAAK,sB;QAAA,SAaKB,G;MACTg,QAAQ,E;MAER,EAAE,G
AAF,IAAS,C;MACT,EAAE,GAFF,IAAS,C;MACT,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MAEd,
OAAO,C;IACX,C;;MAKwF,iBAAY,KAAZ,C;IAAA,C;;MAkHxD,wB;QAAA,WA0W+C,M;;MAzW3E,QAAQ,E;
MAER,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAekC,oB;QAAA,OAAgB,S;MAAW,mB;QAAA,M
AAe,S;MAAW,wB;QAAA,WAqVR,M;;MApV3E,QAAQ,E;MAER,EAAE,MAAF,IAAY,I;MACZ,EAAE,KAAF,I
AAW,G;MACX,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAiC0C,qB;QAAA,QAAiB,E;MAAI,uB;Q
AAA,UAAoB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MA
CpK,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,E
AAE,YAAF,IAaKB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAiCwC,qB;QAAA,QA0Qy
D,Q;;MA1QK,sB;QAAA,SA0QL,Q;;MA1QoE,wB;QAAA,WaiQtF,M;;MAhQ3E,QAAQ,E;MAER,EAAE,OAAF,
IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAyIC,mB;Q
AAA,MA+P0C,Q;;MA/PJ,0B;QAAA,aAAsB,S;MACzF,QAAQ,E;MAER,EAAE,KAAF,IAAW,G;MACX,EAAE,
YAAF,IAaKB,U;MAEIB,OAAO,C;IACX,C;;MAY2C,uB;QAAA,UA4OgC,Q;;MA5OU,qB;QAAA,QA4OV,Q;;M
A3OvE,QAAQ,E;MAER,EAAE,SAAF,IAAe,O;MACf,EAAE,OAAF,IAAa,K;MAEb,OAAO,C;IACX,C;;MAiDm
F,OAAU,S;IAAV,C;;MACI,OAAc,a;IAAd,C;;MACH,OAAW,U;IAAX,C;;MAML,OAAE,E;IAAH,C;;MACA,OA
AQ,O;IAAR,C;;MACG,OAAW,U;IAAX,C;;MAMR,OAAW,U;IAAX,C;;MACF,OAAS,Q;IAAT,C;;MACC,OAA
U,S;IAAV,C;;MAME,OAAW,W;IAAZ,C;;MACD,OAAW,U;IAAX,C;;MACI,OAAe,c;IAAf,C;;MACJ,OAAW,U;I
AAX,C;;MACA,OAAW,U;IAAX,C;;MAMF,OAAS,Q;IAAT,C;;MACD,OAAQ,O;IAAR,C;;MACF,OAAM,K;IA
AN,C;;MACK,OAAW,U;IAAX,C;;MAMC,OAAU,S;IAAV,C;;MACA,OAAU,S;IAAV,C;;MAMU,OAAM,K;IAA
N,C;;MACG,OAAS,Q;IAAT,C;;MACF,OAAM,K;IAAN,C;;MACC,OAAO,M;IAAP,C;;MAMhB,OAAO,M;IAAP,C;;MACC,OAAQ,O;IAAR,
C;;MACC,OAAS,Q;IAAT,C;;MAMC,OAAQ,O;IAAR,C;;MACA,OAAQ,O;IAAR,C;;MACA,OAAQ,O;IAAR,C;;
MAME,OAAQ,O;IAAR,C;;MACF,OAAM,K;IAAN,C;;MACC,OAAO,M;IAAP,C;;MACC,OAAQ,O;IAAR,C;;M
ACC,OAAS,Q;IAAT,C;;MAMG,OAAM,K;IAAN,C;;MACI,OAAU,S;IAAV,C;;MACD,OAAS,Q;IAAT,C;;MACI,
OAAa,Y;IAAb,C;;MACC,OAAc,a;IAAd,C;;MACL,OAAS,Q;IAAT,C;;MAMT,OAAM,K;IAAN,C;;MACA,OAA
M,K;IAAN,C;;MACI,OAAU,S;IAAV,C;;MAMC,OAAO,M;IAAP,C;;MACE,OAAS,Q;IAAT,C;;MAMJ,OAAO,M
;IAAP,C;;MACC,OAAQ,O;IAAR,C;;MAMD,OAAO,M;IAAP,C;;MACO,OAAc,a;IAAd,C;;MACJ,OAAU,S;IAA
V,C;;MAMK,OAAO,M;IAAP,C;;MACG,OAAU,S;IAAV,C;;MAMZ,OAAW,W;IAAZ,C;;MACN,OAAM,K;IAAN
,C;;MACG,OAAS,Q;IAAT,C;;MACF,OAAM,K;IAAN,C;;MAMN,OAAO,M;IAAP,C;;MACO,OAAc,a;IAAd,C;;
MAMJ,OAAU,S;IAAV,C;;MACD,OAAS,Q;IAAT,C;;MAMM,OAAO,M;IAAP,C;;MACE,OAAS,Q;IAAT,C;;MA
MF,OAAO,M;IAAP,C;;MACG,OAAU,S;IAAV,C;;MACD,OAAS,Q;IAAT,C;;MAMa,OAAQ,O;IAAR,C;;MACC,
OAAS,Q;IAAT,C;;MACH,OAAM,K;IAAN,C;;MACI,OAAU,S;IAAV,C;;MAMvB,OAAS,Q;IAAT,C;;MACA,OA
AS,Q;IAAT,C;;MACC,OAAU,S;IAAV,C;;MACA,OAAU,S;IAAV,C;;MElrIB,qB;QAAA,QAaKB,I;MAAM,sB;Q
AAA,SAAmB,I;MAAM,2B;QAAA,cAAwB,I;MAAM,yB;QAAA,YAAsB,I;MAAM,0B;QAAA,aAAuB,I;MAAM,
0B;QAAA,aAAuB,I;MAAM,sB;QAAA,SAAmB,I;MAAM,0B;QAAA,aAAuB,I;MAAM,0B;QAAA,aAAuB,I;MA
AM,gC;QAAA,mBAA6B,I;MAAM,+B;QAAA,kBAA4B,I;MAAM,gC;QAAA,mBAA6B,I;MAAM,uB;QAAA,UA
AoB,I;MAAM,4B;QAAA,eAAyB,I;MAAM,wB;QAAA,WAAqB,I;MAAM,uB;QAAA,UAAoB,I;MACrf,QAAQ,E
;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,IAAmB,W;MACnB,EAAE,W
AAF,IAAiB,S;MACjB,EAAE,YAAF,IAaKB,U;MACIB,EAAE,YAAF,IAaKB,U;MACIB,EAAE,QAAF,IAAc,M;
MACd,EAAE,YAAF,IAaKB,U;MACIB,EAAE,YAAF,IAaKB,U;MACIB,EAAE,kBAAF,IAAwB,gB;MACxB,EA
AE,iBAAF,IAAuB,e;MACvB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,SAAF,IAAe,O;MACf,EAAE,cAAF,IAA
oB,Y;MACpB,EAAE,UAAF,IAAgB,Q;MACHB,EAAE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;;MASDyC,qB;Q
AAA,QAAqB,S;MAAW,sB;QAAA,SAAsB,S;MAAW,2B;QAAA,cAA4B,S;MAAW,yB;QAAA,YAA0B,S;MAA
W,0B;QAAA,aAA6B,S;MAAW,0B;QAAA,aAA6B,S;MAAW,sB;QAAA,SAAuB,S;MAAW,0B;QAAA,aAA0B,S;
MAAW,0B;QAAA,aAA0B,S;MAAW,gC;QAAA,mBAAoC,S;MAAW,+B;QAAA,kBAAmC,S;MAAW,gC;QAAA
,mBAAoC,S;MAAW,uB;QAAA,UAAwB,S;MAAW,4B;QAAA,eAA4B,S;MAAW,wB;QAAA,WAAoB,S;MAAW

,uB;QAAA,UAAmB,S;MACtnB,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,IAAmB,W;MACnB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,QAAF,IAAc,M;MACd,EAAE,YAAF,IAAkB,U;MACIB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,SAAF,IAAe,O;MACf,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;MAYwC,wB;QAAA,WAA4C,S;MAAW,qB;QAAA,QAAiB,S;MAAW,sB;QAAA,SAAkB,S;MAAW,2B;QAAA,cAAuB,S;MAAW,yB;QAAA,YAAqB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,sB;QAAA,SAAkB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,gC;QAAA,mBAA4B,S;MAAW,+B;QAAA,kBAA2B,S;MAAW,gC;QAAA,mBAA4B,S;MAAW,uB;QAAA,UAAmB,S;MAAW,4B;QAAA,eAAwB,S;MAAW,wB;QAAA,WAAoB,S;MAAW,uB;QAAA,UAAmB,S;MAC9IB,QAAQ,E;MAER,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,IAAmB,W;MACnB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,QAAF,IAAc,M;MACd,EAAE,YAAF,IAAkB,U;MACIB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,SAAF,IAAe,O;MACf,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,SAAF,IAAe,O;MAEf,OA AO,C;IACX,C;MAsD0C,qB;QAAA,QAAiB,S;MAAW,sB;QAAA,SAAkB,S;MAAW,2B;QAAA,cAAuB,S;MAAW,yB;QAAA,YAAqB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,sB;QAAA,SAAkB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,gC;QAAA,mBAA4B,S;MAAW,+B;QAAA,kBAA2B,S;MAAW,gC;QAAA,mBAA4B,S;MAAW,uB;QAAA,UAAmB,S;MAAW,4B;QAAA,eAAwB,S;MAAW,wB;QAAA,WAAoB,S;MAAW,uB;QAAA,UAAmB,S;MACziB,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,IAAmB,W;MACnB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,QAAF,IAAc,M;MACd,EAAE,YAAF,IAAkB,U;MACIB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,SAAF,IAAe,O;MACf,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;MAYDqC,qB;QAAA,QAAc,S;MAAW,sB;QAAA,SA Ae,S;MAAW,2B;QAAA,cAAuB,S;MAAW,yB;QAAA,YAAqB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,sB;QAAA,SAAkB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,0B;QAAA,aAAsB,S;MAAW,gC;QAAA,mBAA6B,S;MAAW,+B;QAAA,kBAA4B,S;MAAW,gC;QAAA,mBAA6B,S;MAAW,uB;QAAA,UAAmB,S;MAAW,4B;QAAA,eAAqB,S;MAAW,wB;QAAA,WAAoB,S;MAAW,uB;QAAA,UAAmB,S;MACxhB,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,IAAmB,W;MACnB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,QAAF,IAAc,M;MACd,EAAE,YAAF,IAAkB,U;MACIB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,SAAF,IAAe,O;MACf,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;MAGBsE,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACHJ,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEH B,OAAO,C;IACX,C;MAagD,qB;QAAA,QAAiB,I;MAAM,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACjJ,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEHb,OAAO,C;IACX,C;MAwCyC,qB;QAAA,QAAiB,K;MAAO,qB;QAAA,QAAiB,K;MAC9E,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MAEb,OAAO,C;IACX,C;MAuB8B,mB;QAAA,MAAe,S;MAAW,mB;QAAA,MAAe,S;MACnE,QAAQ,E;MAER,EAAE,KAAF,IAAW,G;MACX,EAAE,KAAF,IAAW,G;MAEX,OAAO,C;IACX,C;MAYuC,qB;QAAA,QAAiB,S;MAAW,qB;QAAA,QAAiB,S;MAAW,mB;QAAA,MAAe,S;MAAW,mB;QAAA,MAAe,S;MACpI,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MACb,EAAE,KAAF,IAAW,G;MACX,EAAE,KAAF,IAAW,G;MAEX,OAAO,C;IACX,C;MAY6B,mB;QAAA,MAAY,S;MAAW,mB;QAAA,MAAY,S;MAC5D,QAAQ,E;MAER,EAAE,KAAF,IAAW,G;MACX,EAAE,KAAF,IAAW,G;MAEX,OAAO,C;IACX,C;MAYsC,qB;QAAA,QAAc,S;MAAW,qB;QAAA,QAAc,S;MAAW,mB;QAAA,MAAY,S;MAAW,mB;QAAA,MAAY,S;MACvH,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MACb,EAAE,KAAF,IAAW,G;MACX,EAAE,K

AAF,IAAW,G;MAEX,OAAO,C;IACX,C;;MAe6C,qB;QAAA,QAAkB,S;MAAW,qB;QAAA,QAAkB,S;MACxF,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MAEb,OAAO,C;IACX,C;;MAe+C,qB;QAAA,QAAiB,S;MAAW,qB;QAAA,QAAiB,S;MACxF,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MAEb,OAAO,C;IACX,C;;MAOI,QAAQ,E;MAGR,OAAO,C;IACX,C;;MAOI,QAAQ,E;MAGR,OAAO,C;IACX,C;;MAOI,QAAQ,E;MAGR,OAAO,C;IACX,C;;MAS8B,wB;QAAA,WAAkC,S;MAC5D,QAAQ,E;MAER,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAMsF,OAAO,M;IAAP,C;;MACC,OAAQ,O;IAAR,C;;MAML,OAAO,M;IAAP,C;;MACO,OAAC,a;IAAd,C;;MACP,OAAO,M;IAAP,C;;MACC,OAAQ,O;IAAR,C;;MAMD,OAAO,M;IAAP,C;;MACU,OAAiB,gB;IAAjB,C;;MAMZ,OAAa,Y;IAAb,C;;MACC,OAAC,a;IAAd,C;;MACD,OAAa,Y;IAAb,C;;MC7oB7C,yB;QAAA,YAAkB,C;MAAG,qB;QAAA,QAAiB,G;MAAK,sB;QAAA,SAAkB,G;MAAK,wB;QAAA,WAAmB,G;MAAI,kC;QAAA,qBAA6B,G;MAAI,qB;QAAA,QAAc,C;MAAG,qB;QAAA,QAAc,C;MAAG,qB;QAAA,QAAc,C;MAAG,2B;QAAA,cAAuB,E;MAAI,yB;QAAA,YAAsB,K;MAAO,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,uB;QAAA,UAAgB,C;MAAG,sB;QAAA,SAAiB,C;MAAG,uB;QAAA,UAAkB,C;MAAG,6B;QAAA,gBAA8B,I;MAAM,uB;QAAA,UAAoB,K;MAAO,wB;QAAA,WAAqB,K;MAAO,sB;QAAA,SAAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,gC;QAAA,mBAA6B,K;MAAO,0B;QAAA,aAAuB,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,+B;QAAA,kBAA4B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,6B;QAAA,gBAA0B,K;MAAO,8B;QAAA,iBAA2B,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,oB;QAAA,OAAgB,I;MAAM,sB;QAAA,SAAE,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC19B,QAAQ,E;MAER,EAAE,WAAF,IAAiB,S;MACjB,EAAE,OAAF,IAAa,K;MACb,EAAE,QAAF,IAAc,M;MACd,EAAE,UAAF,IAAgB,Q;MACHB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MACb,EAAE,aAAF,IAAmB,W;MACnB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,eAAF,IAAqB,a;MACrB,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MACHB,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,eAAF,IAAqB,a;MACrB,EAAE,iBAAF,IAAuB,e;MACvB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,eAAF,IAAqB,a;MACrB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MC/CwC,oB;QAAA,OAAiB,I;MAAM,sB;QAAA,SAAmB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,uB;QAAA,UAAoB,K;MACpI,QAAQ,E;MAER,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;;MAuFuF,iBAAAY,KAAZ,C;IAAA,C;;MACe,UAAAY,KAAZ,IAAqB,O;IAAS,C;;MAGbxC,iBAAAY,KAAZ,C;IAAA,C;;MACc,UAAAY,KAAZ,IAAqB,O;IAAS,C;;MAG5C,iBAAAY,KAAZ,C;IAAA,C;;MACc,UAAAY,KAAZ,IAAqB,O;IAAS,C;;MAGG/C,iBAAAY,KAAZ,C;IAAA,C;;MACc,UAAAY,KAAZ,IAAqB,O;IAAS,C;;MAMnNc,iBAAAY,KAAZ,C;IAAA,C;;MACc,UAAAY,KAAZ,IAAqB,O;IAAS,C;;MAMHpD,iBAAAY,KAAZ,C;IAAA,C;;MACc,UAAAY,KAAZ,IAAqB,O;IAAS,C;;MH/exG,sB;QAAA,SAAkB,S;MAAW,uB;QAAA,UAAmB,S;MAAW,oB;QAAA,OAAgB,S;MAAW,wB;QAAA,WAAoB,S;MAAW,8B;QAAA,iBAA0B,S;MAAW,oB;QAAA,OAAqB,S;MAAW,2B;QAAA,cAAmC,S;MAAW,qB;QAAA,QAAuB,S;MAAW,wB;QAAA,WAA6B,S;MAAW,yB;QAAA,YAAqB,S;MAAW,yB;QAAA,YAAsB,S;MAAW,wB;QAAA,WAAe,S;MAC5Z,QAAQ,E;MAER,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,MAAF,IAAY,I;MACZ,EAAE,UAAF,IAAgB,Q;MACHB,EAAE,gBAAF,IAAsB,c;MACtB,EAAE,MAAF,IAAY,I;MACZ,EAAE,aAAF,IAAmB,W;MACnB,EAAE,OAAF,IAAa,K;MACb,EAAE,UAAF,IAAgB,Q;MACHB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,WAAF,IAAiB,S;MACjB,EAAE,QAAF,IAAc,Q;MAEd,OAAO,C;IACX,C;;MA0C+B,sB;QAAA,SAAiB,G;MAAK,0B;QAAA,aAAsB,I;MAAM,uB;QAAA,UAAmB,S;MACHg,QAAQ,E;MAER,EAAE,QAAF,IAAc,M;MACd,EAAE,YAAF,IAAkB,U;MACIB,EAAE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;;MAMmE,OAAG,E;IAAH,C;;MACA,OAAQ,O;IAAR,C;;MACD,OAAO,M;IAAP,C;;MACC,OAAQ,O;IAAR,C;;MACC,OAAS,Q;IAAT,C;;MACD,OAAQ,O;IAAR,C;;MACA,OAAQ,O;IAAR,C;;MACA,OAAQ,O;IAAR,C;;MAMc,OAAG,E;IAAH,C;;MACG,OAAW,U;IAAX,C;;MACH,OAAQ,O;IAAR,C;;MACD,OAAO,M;IAAP,C;;MACC,OAAQ,O;IAAR,C;;MACG,OAAW,U;IAAX,C;;MACH,OAAQ,O;IAAR,C;;MACC,OAAS

,Q;IAAT,C;;MACA,OAAS,Q;IAAT,C;;MACA,OAAS,Q;IAAT,C;;MACO,OAAgB,e;IAAhB,C;;MACD,OAAe,c;IAAf,C;;MACP,OAAQ,O;IAAR,C;;MACC,OAAS,Q;IAAT,C;;MACF,OAAO,M;IAAP,C;;MAMV,OAAW,U;IAAX,C;;MACG,OAAc,a;IAAd,C;;MACJ,OAAU,S;IAAV,C;;MACH,OAAO,M;IAAP,C;;MAMc,OAAO,M;IAAP,C;;MACO,OAAc,a;IAAd,C;;MACJ,OAAU,S;IAAV,C;;MAMZ,OAAU,S;IAAV,C;;MACC,OAAW,U;IAAX,C;;MACF,OAAAS,Q;IAAT,C;;MACE,OAAW,U;IAAX,C;;MACG,OAAc,a;IAAd,C;;MACG,OAAiB,gB;IAAjB,C;;MAMF,OAAAS,Q;IAAT,C;;MACD,OAAQ,O;IAAR,C;;MACC,OAAS,Q;IAAT,C;;MAMP,OAAQ,O;IAAR,C;;MACD,OAAO,M;IAAP,C;;MACG,OAAU,S;IAAV,C;;MACF,OAAQ,O;IAAR,C;;MACC,OAAS,Q;IAAT,C;;MACQ,OAAiB,gB;IAAjB,C;;MI1N5C,oB;QAAA,OAAgB,E;MAC9C,QAAQ,E;MAER,EAAE,MAAF,IAAY,I;MAEZ,OAAO,C;IACX,C;;MAiBkC,4B;QAAA,eAAqB,S;MAAW,oB;QAAA,OAAgB,E;MAC9E,QAAQ,E;MAER,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,MAAF,IAAY,I;MAEZ,OAAO,C;IACX,C;;MAskF,iBAA,Y,KAAZ,C;IAAA,C;;MC+B5C,mB;QAAA,MAwGuD,M;;MAxGG,oB;QAAA,OAAgB,E;MAAI,oB;QAAA,OAAgB,E;MAAI,mB;QAAA,MAAe,E;MAAI,qB;QAAA,QAAiB,S;MAAW,oB;QAAA,OAAgB,S;MAAW,qB;QAAA,QAAiB,S;MAAW,qB;QAAA,QAAiB,S;MAAW,uB;QAAA,UAAmB,S;MAAW,yB;QAAA,YAAqB,S;MAAW,wB;QAAA,WAAqB,K;MAAO,sB;QAAA,SAAmB,K;MAAO,wB;QAAA,WAAqB,K;MAAO,kC;QAAA,qBAA+B,K;MAAO,sB;QAAA,SAAmB,K;MAAO,oB;QAAA,OAAa,I;MAAM,uB;QAAA,UAAc,E;MAC/gB,QAAQ,E;MAER,EAAE,KAAF,IAAW,G;MACX,EAAE,MAAF,IAAY,I;MACZ,EAAE,MAAF,IAAY,I;MACZ,EAAE,KAAF,IAAW,G;MACX,EAAE,OAAF,IAAa,K;MACb,EAAE,MAAF,IAAY,I;MACZ,EAAE,OAAF,IAAa,K;MACb,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,WAAF,IAAiB,S;MACjB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,QAAF,IAAc,M;MACd,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,oBAAF,IAA0B,kB;MAC1B,EAAE,QAAF,IAAc,M;MACd,EAAE,MAAF,IAAY,I;MACZ,EAAE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;;MAeS,e,oB;QAAA,OAAgB,S;MACIF,QA AQ,E;MAER,EAAE,QAAF,IAAc,M;MACd,EAAE,OAAF,IAAa,K;MACb,EAAE,MAAF,IAAY,I;MAEZ,OAAO,C;IACX,C;;MASyC,mB;QAAA,MAAe,E;MACpD,QAAQ,E;MAER,EAAE,KAAF,IAAW,G;MAEX,OAAO,C;IACX,C;;MAoBqE,sB;QAAA,SAAkB,E;MAAI,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACrK,QAAQ,E;MAER,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAM2F,OAAU,S;IAAV,C;;MACD,OAAS,Q;IAAT,C;;MACC,OAAU,S;IAAV,C;;MAML,OAAO,M;IAAP,C;;MACD,OAAM,K;IAAN,C;;MACA,OAAM,K;IAAN,C;;MC9H/C,qB;QAAA,QAAiB,S;MAAW,oB;QAAA,OP4qIW,S;;MO3qIzE,QAAQ,E;MAER,EAAE,OAAF,IAAa,K;MACb,EAAE,MAAF,IAAY,I;MAEZ,OAAO,C;IACX,C;;MAGcG D,oB;QAAA,OAAa,S;MAAW,sB;QAAA,SAAkB,S;MAAW,2B;QAAA,cAAuB,S;MAAW,sB;QAAA,SAA2C,S;MAAW,qB;QAAA,QAA6B,S;MAAW,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/S,QAAQ,E;MAER,EAAE,MAAF,IAAY,I;MACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,IAAmB,W;MACnB,EAAE,QAAF,IAAc,M;MACd,EAAE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MA0CqC,mC;QAAA,sBAAGC,K;MAAO,oB;QAAA,OAuSD,Q;;MatSvE,QAAQ,E;MAER,EAAE,qBAAF,IAA2B,mB;MAC3B,EAAE,MAAF,IAAY,I;MAEZ,OAAO,C;IACX,C;;MAasC,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAChH,QAAQ,E;MAER,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAoBI,QAAQ,E;MAER,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;;MAyBoD,wB;QAAA,WAAoB,I;MAAM,wB;QAAA,WAAqB,K;MAAO,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACpL,QAAQ,E;MAER,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAkB2D,sB;QAAA,SAAkB,M;MAAQ,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MAC/J,QAAQ,E;MAER,EAAE,SAAF,IAAe,O;MACf,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,OAAO,C;IACX,C;;MAe4D,sB;QAAA,SAAkB,S;MAAW,uB;QAAA,UAA0B,S;MAC/G,QAAQ,E;MAER,EAAE,UAAF,IAAgB,Q;MAChB,EAAE,QAAF,IAAc,M;MACd,EAAE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;;MAGc6C,oB;QAAA,OAAa,S;MAAW,sB;QAAA,SAAkB,S;MAAW,2B;QAAA,cAAuB,S;MAAW,sB;QAAA,SAAmD,S;MAAW,qB;QAAA,QAA6B,S;MAAW,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;QAAA,WAAqB,K;MACpT,QAAQ,E;MAER,EAAE,MAAF,IAAY,I;M

ACZ,EAAE,QAAF,IAAc,M;MACd,EAAE,aAAF,IAAmB,W;MACnB,EAAE,QAAF,IAAc,M;MACd,EAAE,OAA
F,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;MAEhB,
OAAO,C;IACX,C;;MA+BoC,4B;QAAA,eAAyB,K;MAAO,4B;QAAA,eAAyB,K;MAAO,0B;QAAA,aAAuB,K;M
AAO,yB;QAAA,YAAqB,S;MACnJ,QAAQ,E;MAER,EAAE,cAAF,IAAoB,Y;MACpB,EAAE,cAAF,IAAoB,Y;M
ACpB,EAAE,YAAF,IAAkB,U;MACIB,EAAE,WAAF,IAAiB,S;MAEjB,OAAO,C;IACX,C;;MAkBsC,oB;QAAA,
OAAgB,S;MAAW,uB;QAAA,UAAoB,S;MAAW,wB;QAAA,WAAsB,S;MAAW,uB;QAAA,UAA8B,S;MAC3J,Q
AAQ,E;MAER,EAAE,MAAF,IAAY,I;MACZ,EAAE,SAAF,IAAe,O;MACf,EAAE,UAAF,IAAgB,Q;MACHb,EA
AE,SAAF,IAAe,O;MAEf,OAAO,C;IACX,C;;MAuBsF,OAAa,Y;IAAb,C;;MACD,OAAAY,W;IAAZ,C;;MACC,OA
Aa,Y;IAAb,C;;MACD,OAAAY,W;IAAZ,C;;MACA,OAAAY,W;IAAZ,C;;MAMIB,OAAAY,W;IAAZ,C;;MACA,OAA
Y,W;IAAZ,C;;MACH,OAAS,Q;IAAT,C;;MACF,OAAO,M;IAAP,C;;MAMI,OAAS,Q;IAAT,C;;MACA,OAAS,Q;
IAAT,C;;MACM,OAAe,c;IAAf,C;;MACT,OAAM,K;IAAN,C;;MCtX3B,gC;QAAA,mBAA6B,K;MAAO,sB;QAA
A,SA Ae,C;MAAG,qB;QAAA,QAac,C;MAAG,uB;QAAA,UAAoB,K;MAAO,0B;QAAA,aAAuB,K;MAAO,wB;Q
AAA,WAAqB,K;MACrL,QAAQ,E;MAER,EAAE,kBAAF,IAAwB,gB;MACxB,EAAE,QAAF,IAAc,M;MACd,EA
AE,OAAF,IAAa,K;MACb,EAAE,SAAF,IAAe,O;MACf,EAAE,YAAF,IAAkB,U;MACIB,EAAE,UAAF,IAAgB,Q;
MAEhB,OAAO,C;IACX,C;;MAMiG,OAAG,E;IAAH,C;;MACM,OAAc,a;IAAd,C;;MACP,OAAO,M;IAAP,C;;M
ACI,OAAW,U;IAAX,C;;MACJ,OAAO,M;IAAP,C;;MACA,OAAO,M;IAAP,C;;MC7G3D,qB;QAAA,8C;MAAA,k
B;IAAD,C;;MACH,c;MAAA,iB;MAAA,uB;IAAA,C;;MAAA,4C;MAAA,C;MAKI,0E;MAEA,sE;IAPJ,C;;MAKI
,+B;MAAA,0C;IAAA,C;;MAEA,+B;MAAA,wC;IAAA,C;;MAPJ,yF;IAAA,C;;MAAA,a;aAAA,S;UAAA,+C;a
AA,O;UAAA,6C;gBAAA,8D;;IAAA,C;;;MAyBA,8B;IADmC,C;;MAQnC,8B;IADqC,C;;;ICxDzC,C;;;ICMA,C;;;
ICCA,C;;;IA0BA,C;;;ICvBgD,C;;;MAlIb,c;;;Q5Hq8C7C,Q;QADhB,IAAI,mCAAsB,cAA1B,C;UAAqC,aAAO,K;U
AAP,e;;QACrB,sB;QAAhB,OAAGB,cAAhB,C;UAAgB,2B;UAAM,I4Hr8C6C,O5Hq8C/B,S4Hr8C+B,U5Hq8C7C,
C;YAAwB,aAAO,I;YAAP,e;;QAC9C,aAAO,K;;;M4Ht8CsD,iB;IAAA,C;;MAGhD,c;;;Q5H66CG,Q;QADhB,IAAI
,c4H56CA,Q5H46CA,iB4H56CA,Q5H46CsB,UAA1B,C;UAAqC,aAAO,I;UAAP,e;;QACrB,O4H76CZ,Q5H66CY
,W;QAAhB,OAAGB,cAAhB,C;UAAgB,yB;UAAM,IAAI,C4H76CP,oB5H66CkB,O4H76CIB,C5H66CG,C;YAAy
B,aAAO,K;YAAP,e;;;QAC/C,aAAO,I;;;M4H96CH,iB;IAAA,C;;MAE8B,qBAAQ,C;IAAR,C;;MAE6B,qB;QAC3
D,OAAI,OAAO,uBAAX,GAAiB,mBAAjB,GAA6C,SAAH,EAAG,C;MADc,C;IAAA,C;;MAA7B,4BAAa,IAAb,E
AAmB,GAAAnB,EAAwB,GAAxB,kBAA6B,wCAA7B,C;IAAA,C;;MAQU,uBAAGB,IAAhB,C;IAAA,C;;MAMgB,
yBAAGB,IAAhB,EAAsB,KAAtB,C;IAAA,C;;MC/BhE,e;MAAA,iB;MAAA,uB;IAAA,C;;MAAA,+B;MAAA,C;
MACI,4C;MACA,kD;MACA,0C;MACA,8C;IAJJ,C;;;MACI,kB;MAAA,2B;IAAA,C;;;MACA,kB;MAAA,8B;IAA
A,C;;;MACA,kB;MAAA,0B;IAAA,C;;;MACA,kB;MAAA,4B;IAAA,C;;;MAJJ,sH;IAAA,C;;;MAAA,a;aAAA,O;U
AAA,gC;aAAA,U;UAAA,mC;aAAA,M;UAAA,+B;aAAA,Q;UAAA,iC;gBAAA,6D;;IAAA,C;;;MAYI,mD;MACA
,2BAA4B,I;IANhC,C;;MAUqB,IAAN,I;MnIUX,IAAI,EmIXQ,mDnIWR,CAAJ,C;QACI,cAda,qB;QAeb,MAAM,g
CAAyB,OAAQ,WAAjC,C;;MmIZC,QAAM,oBAAN,M;aACH,M;UAAc,Y;UAAAd,K;aACA,O;UAAe,W;UAAf,K;
gBACQ,wC;UAHL,K;;MAAP,W;IAKJ,C;;MAMW,Q;MAHP,IAAI,CAAC,cAAL,C;QAAGB,MAAM,6B;MACTb,
mD;MAEA,OAAO,2F;IACX,C;;MAGI,iD;MACA,kB;MACA,OAAO,kD;IACX,C;;MAkBI,2BAAAY,K;MACZ,gD;
IACJ,C;;MAMI,+C;IACJ,C;;;MCiCA,oC;MAPfGe,6B;IAA1B,C;;MAIC,0C;IAAA,C;;MAEiB,uB;;;Q9HmR3C,Q;Q
ADb,YAAAY,C;QACC,sB;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,I8HpRmE,O9HoRrD,I8HpRqD,U9HoRnE,C;Y
ACI,sBAAO,K;YAAP,wB;;UACJ,qB;;QAEJ,sBAAO,E;;;M8HxRiD,0B;IAAA,C;;MAEL,sB;;;Q9H4S5D,eAAoB,0B
AAa,SAAb,C;QACpB,OAAO,QAAS,cAAhB,C;UACI,I8H9SsE,O9H8SxD,QAAS,W8H9S+C,U9H8StE,C;YACI,q
BAAO,QAAS,Y;YAAhB,uB;;;QAGR,qBAAO,E;;;M8HITqD,yB;IAAA,C;;MAEb,+CAAiB,CAAJB,C;IAAA,C;;M
AEU,+CAAiB,KAAjB,C;IAAA,C;;MAEK,gCAAQ,IAAR,EAAC,SAAd,EAAyB,OAAzB,C;IAAA,C;;MAE8C,uB;
MAA/E,kB;MAAmC,4B;MAC5D,eAAyB,C;MAGrB,+DAAkB,gBAAIB,EAA6B,OAA7B,EAAsC,WAAK,KAA3
C,C;MACA,eAAa,UAAU,gBAAV,I;IALO,C;;MASpB,+DAAkB,KAAIB,EAAyB,YAAzB,C;MAEA,OAAO,wBA
AK,mBAAAY,KAAZ,IAAL,C;IACX,C;;MAE+B,mB;IAAA,C;;;MAS/B,IAAI,UAAU,IAAd,C;QAoB,OAAO,I;M
AC3B,IAAI,2BAAJ,C;QAauB,OAAO,K;MAE9B,OAAO,2DAAC,IAAd,EAAoB,KAApB,C;IACX,C;;MAK+B,oE
AAGB,IAAhB,C;IAAA,C;;MAE/B,oB;MACI,eACsB,C;IAF1B,C;;MAIsC,sBAAQ,gB;IAAR,C;;MAInB,gB;MAD
X,IAAI,CAAC,cAAL,C;QAAGB,MAAM,6B;MACX,iE;MAAX,OAAO,+B;IACX,C;;;MAGJ,oB;MAGwD,iD;MA
GhD,gEAAmB,KAAAnB,EAA0B,WAAkB,KAA5C,C;MACA,eAAa,K;IAJoB,C;;MAOC,sBAAQ,C;IAAR,C;;MAE

N,mB;IAAA,C;;MAG5B,IAAI,CAAC,kBAAL,C;QAAoB,MAAM,6B;MAC1B,OAAO,yBAAI,mCAAJ,EAAI,YA
AJ,E;IACX,C;;MAEoC,sBAAQ,CAAR,I;IAAA,C;;MAGxC,sC;IAAA,C;;MAEQ,IAAI,QAAQ,CAAR,IAAa,SA
S,IAA1B,C;QACI,MAAM,8BAA0B,YAAS,KAAT,gBAAuB,IAAjD,C;;IAEd,C;;MAGI,IAAI,QAAQ,CAAR,IAAa
,QAAQ,IAAzB,C;QACI,MAAM,8BAA0B,YAAS,KAAT,gBAAuB,IAAjD,C;;IAEd,C;;MAGI,IAAI,YAAAY,CAAZ
,IAAiB,UAAU,IAA/B,C;QACI,MAAM,8BAA0B,gBAAa,SAAb,mBAaKc,OAAIC,gBAaKd,IAA5E,C;;MAEV,I
AAI,YAAAY,OAAhB,C;QACI,MAAM,gCAAyB,gBAAa,SAAb,oBAAmC,OAA5D,C;;IAEd,C;;MAGI,IAAI,aAAa,
CAAb,IAAkB,WAAW,IAAjC,C;QACI,MAAM,8BAA0B,iBAAc,UAAo,oBAaQc,QAArC,gBAAsD,IAAhF,C;;M
AEV,IAAI,aAAa,QAAjB,C;QACI,MAAM,gCAAyB,iBAAc,UAAo,qBAAsC,QAA/D,C;;IAEd,C;;MAIc,UACsB,
M;MAFhC,iBAAc,C;MACL,mB;MAAV,OAAU,cAAV,C;QAAU,mB;QACN,aAAW,MAAK,UAAI,SAAiB,6DA
AiB,CAAIC,K;;MAEf,OAAO,U;IACX,C;;MAMiB,Q;MAHb,IAAI,CAAE,KAAF,KAAU,KAAM,KAAPB,C;QAA
0B,OAAO,K;MAEjC,oBAAoB,KAAM,W;MACb,mB;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,gBAAGb,aAAc,O;
QAC9B,IAAI,cAAQ,SAAR,CAAJ,C;UACI,OAAO,K;;MAGf,OAAO,I;IACX,C;;MAIDJ,6C;QAAA,4B;;MAAA,
sC;IAAA,C;;MCsCA,mC;MAzCA,uBAC6B,I;MAMc7B,yBACsC,I;IARHE,C;;MAGpC,OAAO,6BAAc,GAAd,S;I
ACX,C;;MAEwE,gBAAR,Y;MAAQ,c;;Q/H67CxD,Q;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,aAAO,K;UA
AP,e;;QACrB,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,I+H77CwD,O/H67C1C,O+H77C6C,MAAH,Q/
H67CxD,C;YAAwB,aAAO,I;YAAP,e;;QAC9C,aAAO,K;;M+H97CyD,iB;IAAA,C;;MAI5D,IAAI,gCAAJ,C;QA
A+B,OAAO,K;MACtC,UAAU,KAAM,I;MACHB,YAAAY,KAAM,M;M3I8JO,Q;M2I7JzB,e3I6J4C,CAAnB,mDAA
mB,Y2I7JzB,G3I6JyB,C;M2I3J5C,IAAI,eAAS,QAAT,CAAJ,C;QACI,OAAO,K;;MAIP,6B;MAAA,W;Q3IuKqB,
U;Q2IvKD,U3IuKoB,CAAnB,uDAAmB,oB2IvKP,G3IuKO,C;;M2IvK5C,W;QACI,OAAO,K;;MAGX,OAAO,I;IA
CX,C;;MASI,IAAI,UAAU,IAAd,C;QAAoB,OAAO,I;MAC3B,IAAI,0BAAJ,C;QAAyB,OAAO,K;MACHc,IAAI,c
AAQ,KAAM,KAAIB,C;QAAwB,OAAO,K;MAEV,gBAAd,KAAM,Q;MAAQ,c;;Q/Hw4CT,Q;QADhB,IAAI,wCA
AsB,mBAA1B,C;UAAqC,aAAO,I;UAAP,e;;QACrB,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,C+
Hx4CK,2B/Hw4CM,O+Hx4CN,C/Hw4CT,C;YAAyB,aAAO,K;YAAP,e;;QAC/C,aAAO,I;;M+Hz4CH,iB;IACJ,C
;;MAEwC,Q;MAAA,4CAAc,GAAd,8B;IAAA,C;;MAQT,OAAQ,SAAR,YAAQ,C;IAAR,C;;MAEG,qBAAQ,C;IA
AR,C;;MACH,OAAA,YAAQ,K;IAAR,C;;MAWX,wC;MAAS,sB;IAAT,C;;MACkD,+CAAY,OAAZ,C;IAAA,C;;
MAI3C,kD;IAAA,C;;MAC+B,OAAA,0BAAc,U;IAAd,C;;MACT,OAAA,0BAAc,OAAO,I;IAArB,C;;MAH7B,oB
AAoB,6BAAQ,W;MAC5B,+F;IAIJ,C;;MAE+B,OAAA,qBAAiB,K;IAAjB,C;;MAZvC,IAAI,4BAAJ,C;QACI,+E;;
MAcJ,OAAO,mC;IACX,C;;MAMmE,qB;QAAE,2CAAS,EAAT,C;MAAF,C;IAAA,C;;MAArC,OAAQ,eAAR,YA
AQ,EAAa,IAAb,EAAmB,GAAnB,EAAwB,GAAxB,kBAA6B,iCAA7B,C;IAAR,C;;MAEqB,+BAAS,KAAM,IAA
f,IAAsB,GAAtB,GAA4B,wBAAS,KAAM,MAAf,C;IAA5B,C;;MAEf,OAAI,MAAM,IAAV,GAAGB,YAAhB,GA
AoC,SAAF,CAAE,C;IAApC,C;;MAWIB,wC;MAAS,6B;IAAT,C;;MACgE,iDAAc,OAAoC,IAAA,C;;MAI3D,kD;
IAAA,C;;MAC+B,OAAA,0BAAc,U;IAAd,C;;MACT,OAAA,0BAAc,OAAO,M;IAArB,C;;MAH7B,oBAAoB,6B
AAQ,W;MAC5B,iG;IAIJ,C;;MAE+B,OAAA,qBAAiB,K;IAAjB,C;;MAZvC,IAAI,8BAAJ,C;QACI,mF;;MAcJ,O
AAO,qC;IACX,C;;MAK0D,gBAAR,Y;MAAQ,sB;;Q/HoG9C,Q;QAAA,2B;QAAhB,OAAgB,cAAhB,C;UAAgB,y
B;UAAM,I+HpGsD,O/HoGxC,O+HpG2C,IAAH,M/HoGtD,C;YAAwB,qBAAO,O;YAAP,uB;;QAC9C,qBAAO,I;
;;M+HrG+C,yB;IAAA,C;;MAEtD,qC;IAAA,C;;MAEqE,gC;MAAX,OAAU,CAAC,kBAAN,CAAM,0DAAmB,CA
ApB,KAA4B,oBAAjC,CAAiC,8DAAqB,CAAjD,C;IAAV,C;;MACG,OAAU,SAAL,CAAO,IAAF,mBAAL,CAAY
,MAAP,C;IAAV,C;;MAErD,IAAI,gCAAJ,C;QAA+B,OAAO,K;MACtC,OAAO,OAAA,CAAE,IAAF,EAAS,KAA
M,IAAf,KAAAsB,OAAA,CAAE,MAAF,EAAW,KAAM,MAAjB,C;IACjC,C;;MAPJ,4C;QAAA,2B;;MAAA,qC;I
AAA,C;;MC7GA,mC;MAIB+D,6B;IAA1B,C;;MAQjC,IAAI,UAAU,IAAd,C;QAAoB,OAAO,I;MAC3B,IAAI,0B
AAJ,C;QAAsB,OAAO,K;MAC7B,OAAO,sDAAU,IAAV,EAAGB,KAAhB,C;IACX,C;;MAK+B,qEAAkB,IAAIB,
C;IAAA,C;;MAE/B,qC;IAAA,C;;MAGwB,Q;MADhB,iBAAc,C;MACC,mB;MAAhB,OAAgB,cAAhB,C;QAAgB,
yB;QACC,U;QAAb,2BAAa,yEAAuB,CAApC,K;;MAEJ,OAAO,U;IACX,C;;MAGI,IAAI,CAAE,KAAF,KAAU,K
AAM,KAAPB,C;QAA0B,OAAO,K;MACjC,OAAO,CAAE,qBAAy,KAAZ,C;IACb,C;;MAZJ,4C;QAAA,2B;;M
AAA,qC;IAAA,C;;MCZgB,Q;MpJ+2YA,U;MADhB,UAAe,C;MACf,uD;QAAgB,cAAhB,iB;QACI,YAAgB,OoJj
3YiB,OpJi3YjC,I;;MoJj3Yj,aAAa,iBpJm3YN,GoJn3YM,C;MACb,wBAAGB,SAAhB,gB;QAAgB,gBAAA,SAAh
B,M;QACW,SAAP,MAAO,EAAO,SAAP,C;;MAEX,OAAO,M;IACX,C;;MAWiB,Q;MAFb,YAAAY,iBAAa,gBAA
b,C;MACZ,YAAAY,iBAAa,gBAAb,C;MACZ,wBAAa,SAAb,gB;QAAa,WAAA,SAAb,M;QACI,KAAM,WAAI,IA

AK,MAAT,C;QACN,KAAM,WAAI,IAAK,OAAT,C;;MAEV,OAAO,UAAS,KAAT,C;IACX,C;;MAaW,4B;MAA
A,U;QAAqB,OAAL,SpJszMhB,YAAQ,C;;MoJtzMf,W;IACJ,C;;MAYI,OpJyyMO,qBAAQ,CoJzyMf,GAAe,cAAf,
GAAmC,S;IAAnC,C;;MASA,IAAI,cAAS,KAAb,C;QAAoB,OAAO,I;MAC3B,IAAI,SAAK,OAAL,KAAa,KAAM,
OAAvB,C;QAA6B,OAAO,K;MAEpC,4C;QACI,SAAS,UAAK,CAAL,C;QACT,SAAS,MAAM,CAAN,C;QAET,I
AAI,OAAO,EAAX,C;UACI,Q;;cACG,IAAI,cAAc,UAAIB,C;UACH,OAAO,K;;QAIP,0BAAsB,kBAAtB,C;UAA4
C,IAAI,CAAI,kBAAH,EAAG,EAakB,EAAIB,CAAR,C;YAA+B,OAAO,K;;cACIF,8BAAsB,sBAAtB,C;UAA4C,I
AAI,CAAI,cAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;;cAC9E,+BAAsB,uBAAtB,C;UAA4C,IAAI,C
AAI,cAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;;cAC9E,6BAAsB,qBAAtB,C;UAA4C,IAAI,CAAI,c
AAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;;cAC9E,8BAAsB,sBAAtB,C;UAA4C,IAAI,CAAI,cAAH,
EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;;cAC9E,+BAAsB,uBAAtB,C;UAA4C,IAAI,CAAI,cAAH,EAAG
,EAAc,EAAd,CAAR,C;YAA2B,OAAO,K;;cAC9E,gCAAsB,wBAAtB,C;UAA4C,IAAI,CAAI,cAAH,EAAG,EAA
c,EAAd,CAAR,C;YAA2B,OAAO,K;;cAC9E,8BAAsB,sBAAtB,C;UAA4C,IAAI,CAAI,cAAH,EAAG,EAAc,EAA
d,CAAR,C;YAA2B,OAAO,K;;cAC9E,iCAAsB,yBAAtB,C;UAA4C,IAAI,CAAI,cAAH,EAAG,EAAc,EAAd,CAA
R,C;YAA2B,OAAO,K;;cAE9E,qCAAsB,6BAAtB,C;UAA4C,IAAI,CAAI,gBAAH,EAAG,EAAc,EAAd,CAAR,C;
YAA2B,OAAO,K;;cAC9E,sCAAsB,8BAAtB,C;UAA4C,IAAI,CAAI,gBAAH,EAAG,EAAc,EAAd,CAAR,C;YAA
2B,OAAO,K;;cAC9E,oCAAsB,4BAAtB,C;UAA4C,IAAI,CAAI,gBAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,O
AAO,K;;cAC9E,qCAAsB,6BAAtB,C;UAA4C,IAAI,CAAI,gBAAH,EAAG,EAAc,EAAd,CAAR,C;YAA2B,OAA
O,K;;cAEtE,IAAI,YAAM,EAAN,CAAJ,C;UAAc,OAAO,K;;MAIrC,OAAO,I;IACX,C;;MAOI,aAAa,CAAK,eAAL
,gBAAK,EAAa,SAAb,CAAL,GAA6C,CAA7C,QAAiD,CAAjD,I;MrD3FW,kBAAxB,mBqD4FmB,MrD5FnB,C;M
qd6FI,oD9I7BgD,gB8I6BhD,C;MADJ,OnIICO,W8E1DsC,W;IqD+FjD,C;;MAII,IAAY,SAAR,0BAAJ,C;QACI,M
AAO,gBAAO,OAAP,C;QACP,M;;MAEJ,SAAU,WAAI,SAAJ,C;MACV,MAAO,gBAAO,EAAP,C;MAEP,4C;QA
CI,IAAI,MAAK,CAAT,C;UACI,MAAO,gBAAO,IAAP,C;;QAEX,cAAc,UAAK,CAAL,C;QAEV,IADE,OACF,S;
UAAmB,MAAO,gBAAO,MAAP,C;aAC1B,mBAFE,OAfE,E;UAA2B,4BAAR,OAAQ,EAA4B,MAA5B,EAAoC,
SAApC,C;aAC3B,uBAHE,OAGF,E;UAAmB,MAAO,gBA Ae,cAAR,OAAQ,CAAf,C;aAC1B,wBAJE,OAfE,E;UA
AmB,MAAO,gBA Ae,cAAR,OAAQ,CAAf,C;aAC1B,sBALE,OAKF,E;UAAmB,MAAO,gBA Ae,cAAR,OAAQ,CA
Af,C;aAC1B,uBANE,OAMF,E;UAAmB,MAAO,gBA Ae,cAAR,OAAQ,CAAf,C;aAC1B,wBAPE,OAOF,E;UAAm
B,MAAO,gBA Ae,cAAR,OAAQ,CAAf,C;aAC1B,yBARE,OAQF,E;UAAmB,MAAO,gBA Ae,cAAR,OAAQ,CAAf,
C;aAC1B,uBATE,OASF,E;UAAmB,MAAO,gBA Ae,cAAR,OAAQ,CAAf,C;aAC1B,0BAVE,OAUF,E;UAAmB,M
AAO,gBA Ae,cAAR,OAAQ,CAAf,C;aAE1B,kBAZE,OAYF,c;UAAmB,MAAO,gBA Ae,kBAAR,OAAQ,CAAf,C;a
AC1B,kBAbE,OAfE,e;UAAmB,MAAO,gBA Ae,kBAAR,OAAQ,CAAf,C;aAC1B,kBAfE,OAfE,c;UAAmB,MAAO,gBA Ae,kBAAR,OAAQ,CAAf,C;;U
AEP,MAAO,gBAAO,OAAQ,WAAf,C;;MAIIC,MAAO,gBAAO,EAAP,C;MACP,SAAU,kBAAmB,iBAAV,SAAU
,CAAnB,C;IACd,C;;M9InJA,6B;IAAA,C;;MACsC,Y;IAAA,C;;MACI,Y;IAAA,C;;MACN,Q;IAAA,C;;MACI,S;IA
AA,C;;MACL,MAAM,6B;IAAN,C;;MACI,MAAM,6B;IAAN,C;;;MANvC,oC;QAAA,mB;;MAAA,6B;IAAA,C;;
MASA,yB;MACI,+C;IADJ,C;;MAGgD,qCAAoB,KAAM,U;IAA1B,C;;MACb,Q;IAAA,C;;MACG,W;IAAA,C;;M
AEH,Q;IAAA,C;;MACG,W;IAAA,C;;MACiB,Y;IAAA,C;;MACgB,OAAA,QAAS,U;IAAT,C;;MAE3B,MAAM,8
BAA0B,iDAA8C,KAA9C,MAA1B,C;IAAN,C;;MACM,S;IAAA,C;;MACI,S;IAAA,C;;MAEL,kC;IAAA,C;;MAC
Q,kC;IAAA,C;;MAEjD,IAAI,UAAS,CAAb,C;QAAgB,MAAM,8BAA0B,YAAS,KAAnC,C;MACTb,OAAO,2B;IA
CX,C;;MAGI,IAAI,cAAa,CAAb,IAAkB,YAAW,CAAjC,C;QAAoC,OAAO,I;MAC3C,MAAM,8BAA0B,gBAAa,S
AAb,mBAAkC,OAA5D,C;IACV,C;;MAEiC,8B;IAAA,C;;;MA5BrC,gC;QAAA,e;;MAAA,yB;IAAA,C;;MA+B8D
,6BAAkB,SAAIb,EAAoC,KAApC,C;IAAA,C;;MAE3B,oB;MAA0B,0B;IAA3B,C;;MACC,OAAA,WAAO,O;IAA
P,C;;MACG,OAAA,WNmzM3B,YAAQ,C;IMnzMmB,C;;MACW,OAAO,SAAP,WAAO,EAAS,OAAT,C;IAAP,C
;;MACyB,c;;Qaw4CtD,Q;QADhB,IAAI,cbv4CyD,Qau4CzD,iBbv4CyD,Qau4CnC,UAA1B,C;UAAqC,aAAO,I;UA
AP,e;;QACrB,Obx4C6C,Qaw4C7C,W;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UAAM,IAAI,Cbx4CkD,oBaw4Cv
C,Obx4CuC,Caw4CtD,C;YAAyB,aAAO,K;YAAP,e;;;QAC/C,aAAO,I;;Mbz4CsD,iB;IAAA,C;;MACTb,OAAO,qB
AAP,WAAO,C;IAAP,C;;MAEQ,gBAAP,W;MAAA,OAAwB,cAAxB,GcqGpC,SdrGoC,GcuGpC,SNwkBoB,Q;IR/
qBgB,C;;MAON,8B;IAAA,C;;MAMe,OAAI,QAAS,OAAT,GAAgB,CAApB,GAAgC,OAAT,QAAS,CAAhC,GA
A8C,W;IAA9C,C;mFAErD,yB;MAAA,qD;MAAA,mB;QAK0C,kB;MAAA,C;KAL1C,C;+FAOA,yB;MAAA,+D;

MAAA,mB;QAMwD,uB;MAAA,C;KANxD,C;2FAQA,yB;MAAA,+D;MAAA,mB;QAMoD,uB;MAAA,C;KANp
D,C;;MAaI,OAAI,QAAS,OAAT,KAAiB,CAArB,GAAwB,gBAAXB,GAAyC,iBAAU,sBA
AkB,QAAlB,EAAwC,IAAxC,CAAV,C;IAAzC,C;;MAOA,OAAI,QAAS,OAAT,KAAiB,CAArB,GAAwB,gBAAXB,GAAyC,iBAAU,sBA
AkB,QAAlB,EAAwC,IAAxC,CAAV,C;IAAzC,C;;MAMuD,OAAI,eAAJ,GAAqB,OAAO,OAAP,CAArB,GAA0C,
W;IAA1C,C;;MAMQ,OAAS,cAAT,QAAS,C;IAAT,C;gFAEnE,yB;MAaA,gE;MAbA,6B;QAYBI,WAAW,eAduE,I
AcvE,C;QWDX,iBAAc,CAAd,UXbkF,IWalf,U;UXE6B,eAf2D,IAevD,CWDtB,KXCsB,CAAJ,C;;QAFyC,OAgB/
D,I;MAhB+D,C;KAXIE,C;8FAaA,yB;MAAA,gE;MAAA,6B;QAYI,WAAW,eAAa,IAAb,C;QWDX,iBAAc,CAA
d,UXEO,IWFP,U;UXE6B,eAAI,KWDtB,KXCsB,CAAJ,C;;QAC7B,OAAO,I;MACX,C;KAFa,C;;MAsBY,uBAAG
,iBAAO,CAAP,IAAH,C;IAAA,C;;MAQA,OAAA,SAAK,KAAL,GAAy,CAAZ,I;IAAA,C;;MAOGD,QAAC,mB;I
AAD,C;;MAaxD,OAAO,qBAAGB,SAAK,U;IAChC,C;sFAEA,yB;MAAA,qD;MAAA,4B;QAKgE,uCAAQ,W;MA
AR,C;KALhE,C;sFAOA,yB;MAAA,qD;MAAA,4B;QAKoD,uCAAQ,W;MAAR,C;KALpD,C;;MAGBI,OAAI,mB
AAJ,GAAe,cAAf,GAAmC,S;IAAnC,C;;MAWgH,OAAA,SAAK,qBAAY,QAAZ,C;IAAL,C;;MAEIE,QAAM,cAA
N,C;aAC9C,C;UAD8C,OACzC,W;aACL,C;UAF8C,OAEzC,OAAO,sBAAK,CAAL,CAAP,C;gBAFyC,OAGtC,S;;
IAHsC,C;;MAsBgB,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,c;MACjG,WAAW,cAAX,EAAiB,SAAjB,EA
A4B,OAA5B,C;MAEA,UAAU,S;MACV,WAAW,UAAU,CAAV,I;MAEX,OAAO,OAAO,IAAd,C;QACI,UAAW,
GAAy,GAAN,IAAM,KAAK,C;QAC5B,aAAa,sBAAI,GAAJ,C;QACb,UAAU,cAAc,MAAd,EAAsB,OAAtB,C;Q
AEV,IAAI,MAAM,CAAV,C;UACI,MAAM,MAAM,CAAN,I;aACL,IAAI,MAAM,CAAV,C;UACD,OAAO,MAA
M,CAAN,I;UAEP,OAAO,G;;MAEf,OAAO,EAAE,MAAM,CAAN,IAAF,K;IACX,C;;MAiB8E,yB;QAAA,YAAi
B,C;MAAG,uB;QAAA,UAAe,c;MAC7G,WAAW,cAAX,EAAiB,SAAjB,EAA4B,OAA5B,C;MAEA,UAAU,S;MA
CV,WAAW,UAAU,CAAV,I;MAEX,OAAO,OAAO,IAAd,C;QACI,UAAW,GAAy,GAAN,IAAM,KAAK,C;QAC
5B,aAAa,sBAAI,GAAJ,C;QACb,UAAU,UAAW,SAAQ,MAAR,EAAgB,OAAhB,C;QAErB,IAAI,MAAM,CAAV,
C;UACI,MAAM,MAAM,CAAN,I;aACL,IAAI,MAAM,CAAV,C;UACD,OAAO,MAAM,CAAN,I;UAEP,OAAO,
G;;MAEf,OAAO,EAAE,MAAM,CAAN,IAAF,K;IACX,C;kGAEa,yB;MAAA,8D;MAAA,4D;;QAsBqC,qB;UAA
E,qBAAc,iBAAS,EAAT,CAAd,EAA4B,WAA5B,C;QAAF,C;MAAA,C;MatBrC,+D;QAKBI,yB;UAAA,YAAiB,C
;QACjB,uB;UAAA,UAAe,c;QAGf,+BAAa,SAAb,EAAwB,OAAxB,EAAiC,oCAAjC,C;MAAA,C;KAtBJ,C;;MA4
CoC,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,c;MACnE,WAAW,cAAX,EAAiB,SAAjB,EAA4B,OAA5B,C
;MAEA,UAAU,S;MACV,WAAW,UAAU,CAAV,I;MAEX,OAAO,OAAO,IAAd,C;QACI,UAAW,GAAy,GAAN,I
AAM,KAAK,C;QAC5B,aAAa,sBAAI,GAAJ,C;QACb,UAAU,WAAW,MAAX,C;QAEV,IAAI,MAAM,CAAV,C;
UACI,MAAM,MAAM,CAAN,I;aACL,IAAI,MAAM,CAAV,C;UACD,OAAO,MAAM,CAAN,I;UAEP,OAAO,G;
;MAEf,OAAO,EAAE,MAAM,CAAN,IAAF,K;IACX,C;;MAQQ,gBAAY,OAAZ,C;QAAuB,MAAM,gCAAYB,gB
AAa,SAAb,mCAAKD,OAAID,OAAzB,C;WAC7B,gBAAY,CAAZ,C;QAAiB,MAAM,8BAA0B,gBAAa,SAAb,yB
AA1B,C;WACvB,cAAU,IAAV,C;QAAkB,MAAM,8BAA0B,cAAW,OAAx,gCAA2C,IAA3C,OAA1B,C;IAEhC,
C;;MAcoC,MAAM,wBAAoB,8BAApB,C;IAAoD,C;;MAI1D,MAAM,wBAAoB,8BAApB,C;IAAoD,C;;;wFiG9W
9F,yB;MhG+BA,wE;MgG/BA,uC;QAmBW,kBhGoBiD,oB;QgGO9C,Q;QAAA,OAAK,0B;QAAf,OAAU,cAAV,C
;UAAU,mB;UACN,UAAU,sBAAM,CAAN,C;UACV,kBAAkB,sBAAY,GAAZ,C;UACIB,WhGmHJ,agGnHgB,Gh
GmHhB,EgGjJyC,SA8BIB,CAAU,GAAV,EAAe,WAAf,EAA4B,CAA5B,EAA+B,uBAAuB,CAAC,WAAy,mBA
AY,GAAZ,CAAnE,ChGmHvB,C;;QgGjJA,OAgCO,W;MA/BX,C;KApBA,C;;MA8Cc,Q;MAAA,OAAA,SAAK,iB
;MAAf,OAAU,cAAV,C;QAAU,mB;QACN,UAAU,sBAAM,CAAN,C;QACV,kBAAkB,sBAAY,GAAZ,C;QACIB,
WhGmHJ,agGnHgB,GhGmHhB,EgGnHuB,UAAU,GAAV,EAAe,WAAf,EAA4B,CAA5B,EAA+B,uBAAuB,CAA
C,WAAy,mBAAY,GAAZ,CAAnE,ChGmHvB,C;;MgGjHA,OAAO,W;IACX,C;iFAEA,yB;MAAA,gB;MAAA,8B
;MhGvBA,wE;MgGuBA,6D;QAnCW,kBhGoBiD,oB;QgGO9C,Q;QAAA,OAAK,0B;QAAf,OAAU,cAAV,C;UAA
U,mB;UACN,UAAU,sBAAM,CAAN,C;UACV,kBAAkB,sBAAY,GAAZ,C;UA8BwE,U;UA7B1F,WhGmHJ,agGn
HgB,GhGmHhB,EgGtFkC,UA7BD,GA6BC,EA7BoB,uBAAuB,CAAC,WAAy,mBAAY,GAAZ,CA6BzC,GAAW,
qBA7B3B,GA6B2B,EA7BT,CA6BS,CAAX,GAA6C,UA7BxD,WA6BwD,6DAA5D,EA7BiB,CA6BjB,ChGsFIC,C
;;QgGvFA,OA1BO,W;MA0BP,C;KAvBJ,C;kFA0BA,yB;MAAA,gB;MAAA,8B;MAAA,0E;QAICc,Q;QAAA,OA
AK,0B;QAAf,OAAU,cAAV,C;UAAU,mB;UACN,UAAU,sBAAM,CAAN,C;UACV,kBA6DQ,WA7DU,WAAy,G
AAZ,C;UA6DuF,U;UAAjG,WhGuDZ,agGnHgB,GhGmHhB,EgGvDiD,UA5DhB,GA4DgB,EA5DK,uBAAuB,CA
4DjE,WA5D8E,mBAAY,GAAZ,CA4D1B,GAAW,qBA5D1C,GA4D0C,EA5DxB,CA4DwB,CAAX,GAA6C,UA5

DvE,WA4DuE,6DAA5D,EA5DE,CA4DF,ChGuDjD,C;;QgGxDA,OACY,W;MADZ,C;KA5BJ,C;iFAgCA,yB;MAAA,gB;MAAA,8B;MhGjFA,wE;MgGiFA,qD;QA7FW,kBhGoBiD,oB;QgGO9C,Q;QAAA,OAAK,0B;QAAf,OAAU,cAAV,C;UAAU,mB;UACN,UAAU,sBAAM,CAAN,C;UACV,kBAaKb,sBAAY,GAAZ,C;UAKFiD,U;UAjFnE,WhGmHJ,agGnHgB,GhGmHhB,EgGlCgC,UAjFsB,uBAAuB,CAAC,WAAY,mBAAY,GAAZ,CAiFhD,kBAA6B,UAjFjC,WaIFiC,6DAAvC,EAjFmB,CAiFnB,ChGkChC,C;;QgGnCA,OA9EO,W;MA8EP,C;KAjBJ,C;oFAoBA,yB;MAAA,gB;MAAA,8B;MAAA,KE;QAtFc,Q;QAAA,OAAK,0B;QAAf,OAAU,cAAV,C;UAAU,mB;UACN,UAAU,sBAAM,CAAN,C;UACV,kBA2GQ,WA3GU,WAAY,GAAZ,C;UA2GgE,U;UAA1E,WhGSZ,agGnHgB,GhGmHhB,EgGT+C,UA1GO,uBAAuB,CA0GjE,WA1G8E,mBAAY,GAAZ,CA0GjC,kBAA6B,UA1GhD,WA0GgD,6DAAvC,EA1GI,CA0GJ,ChGS/C,C;;QgGVA,OACY,W;MADZ,C;KAiBJ,C;qFA0BA,yB;MAAA,gB;MAAA,8B;MhG/HA,wE;MgG+HA,uC;QA3IW,kBhGoBiD,oB;QgGO9C,Q;QAAA,OAAK,0B;QAAf,OAAU,cAAV,C;UAAU,mB;UACN,UAAU,sBAAM,CAAN,C;UACV,kBAaKb,sBAAY,GAAZ,C;UACC,oB;UAKIc,U;UAAjC,IAIikD,uBAAuB,CAAC,WAAY,mBAAY,GAAZ,CAkItF,C;YADA,mBAjI+C,C;;YAiI/C,mBACkB,UAlIW,GAKIX,EAAe,UAlIC,WakID,6DAAf,EAlI6B,CAkI7B,C;;UAlIIB,WhGmHJ,agGnHgB,GhGmHhB,mB;;QgGaA,OA9HO,W;MA8HP,C;KAnBJ,C;sFAwBA,yB;MAAA,gB;MAAA,8B;MAAA,oD;QAxIc,Q;QAAA,OAAK,0B;QAAf,OAAU,cAAV,C;UAAU,mB;UACN,UAAU,sBAAM,CAAN,C;UACV,kBA6JQ,WA7JU,WAAY,GAAZ,C;UACC,oB;UA8Jc,U;UAAjC,IA9JkD,uBAAuB,CA4JjE,WA5J8E,mBAAY,GAAZ,CA8JtF,C;YADA,mBA7J+C,C;;YA6J/C,mBACkB,UAA9JW,GA8JX,EAAe,UA9JC,WA8JD,6DAAf,EA9J6B,CA8J7B,C;;UAFV,WhGzCZ,agGnHgB,GhGmHhB,mB;;QgGyCA,OAAy,W;MAAZ,C;KAvBJ,C;;MAxIc,Q;MAAA,OAAK,0B;MAAf,OAAU,cAAV,C;QAAU,mB;QACN,UAAU,sBAAM,CAAN,C;QACV,kBA+KG,WA/Ke,WAAY,GAAZ,C;QA2GgE,U;QAoE/E,WhG3DP,agGnHgB,GhGmHhB,EgG2DmC,CA9KmB,uBAAuB,CA8KtE,WA9KmF,mBAAY,GAAZ,CA0GjC,GAoErC,CAPeQc,GAA6B,UA1GhD,WA0GgD,6DAoEnD,IAAM,CAAN,IhG3DnC,C;;MgG2DA,OAAO,W;IAAP,C;;M8CnPc,kB;MAAuB,kB;IAAxB,C;;MAAC,iB;IANtC,C;;MAM6D,iB;IAN7D,C;;MAAA,wBAMsC,qCANtC,EAM6D,qCAN7D,C;IAAA,C;;MAAA,OAMsC,mDANtC,IAM6D,wCAN7D,O;IAAA,C;;MAAA,c;MAMsC,sD;MAAuB,sD;MAN7D,a;IAAA,C;;MAAA,4IAMsC,sCANtC,IAM6D,sCAN7D,I;IAAA,C;wF1IEA,yB;MAAA,kC;MAAA,4C;;QAMuF,wC;MANvF,C;MAAA,wD;QAO2C,8B;MAAA,C;MAP3C,8E;MAAA,2B;QAMuF,2C;MAAA,C;KANvF,C;;MAcuC,wC;IAAD,C;;MACmB,4BAAiB,wBAAjB,C;IAAA,C;;MAQG,OAAI,oCAAJ,GAA2B,SAAK,KAaHc,GAA0C,I;IAA1C,C;;MAMc,OAAI,oCAAJ,GAA2B,SAAK,KAaHc,GAA0C,S;IAA1C,C;;MAGrB,wBAAO,CAAP,IAAY,mC;IAAZ,C;;MAK7C,kBADE,SACF,O;QADJ,OACc,S;WACV,kBAFE,SAEF,c;QAEQ,yCAAwB,MAAO,KAAP,GAAc,CAAtC,C;UAJZ,OAIuD,S;;UAJvD,OAK6B,mBAAL,SAAK,CAAT,GAA+B,sBAA/B,GAAgD,S;;QALpE,OAoY,sB;IAPZ,C;;MAAI,kBADE,SACF,O;QADJ,OACc,S;WACV,kBAFE,SAEF,c;QAFJ,OAe8B,mBAAL,SAAK,CAAT,GAA+B,sBAA/B,GAAgD,S;;QAFrE,OAGY,sB;IAHZ,C;;MAagB,Q;MADhB,aAAa,gB;MACG,2B;MAAhB,OAAGB,cAAhB,C;QAAGB,yB;QACL,OAAP,MAAO,EAAO,OAAP,C;;MAEX,OAAO,M;IACX,C;;MAYiB,Q;MAHb,mBAAmB,mCAAwB,EAAxB,C;MACnB,YAAY,iBAAa,YAAb,C;MACZ,YAAY,iBAAa,YAAb,C;MACC,2B;MAAAb,OAAa,cAAb,C;QAAa,sB;QACT,KAAM,WAAl,IAAK,MAAT,C;QACN,KAAM,WAAl,IAAK,OAAT,C;;MAEV,OAAO,UAAS,KAAT,C;IACX,C;;MU3EqE,gB;IAAA,C;;MAQD,4BAAiB,SAAjB,C;IAAA,C;;MAOhE,OAAGB,mBAAhB,C;QAAGB,8B;QAAM,UAAU,OAAV,C;;IAC1B,C;;MAKuC,0B;MACnC,eAAoB,C;IADc,C;;MAEM,OAAA,eAAS,U;IAAT,C;;MACqC,Q;MAAhC,wBAAa,oBAAmB,mBAAnB,EAAmB,2BAAnB,QAAb,EAA0C,eAAS,OAAAnD,C;IAAA,C;;sFiI5BjD,yB;MAAA,4E;MAAA,gB;MAAA,8B;MAAA,+C;QAUiC,Q;QAA7B,OAA6B,wCAAqB,QAAS,aAA9B,0D;MAA7B,C;KAVJ,C;wFAYA,yB;MAAA,4E;MAAA,gB;MAAA,8B;MAAA,+C;QAWiC,Q;QAA7B,OAA6B,wCAAqB,QAAS,aAA9B,0D;MAA7B,C;KAXJ,C;wFAaA,yB;MAAA,4E;MAAA,gB;MAAA,8B;MAAA,+C;QAKiC,Q;QAA7B,OAA6B,wCAAqB,QAAS,aAA9B,0D;MAA7B,C;KALJ,C;;MAeI,SAAK,aAAI,QAAS,aAAb,EAAmB,KAAmB,C;IACT,C;;M/IzCA,wB;MACI,8C;IADJ,C;;MAGgD,oCAAsB,KAAM,U;IAA5B,C;;MACb,Q;IAAA,C;;MACG,W;IAAA,C;;MAEH,Q;IAAA,C;;MACG,W;IAAA,C;;MAEa,Y;IAAA,C;;MACO,Y;IAAA,C;;MACd,W;IAAA,C;;MACoB,6B;IAAA,C;;MACvB,6B;IAAA,C;;MACY,8B;IAAA,C;;MAEHb,6B;IAAA,C;;MAjBrC,+B;QAAA,c;;MAAA,wB;IAAA,C;;MA0BuE,Q;MAA7B,OAA6B,uE;IAA7B,C;;MAetC,OAAI,KAAM,OAAN,GAAa,CAAjB,GAA0B,QAAN,KAAM,EAAM,qBAAc,YAAY,KAAM,OAAIB,CAAd,CAAN,CAA1B,GAA6E,U;IAA7E,C;kFAEJ,yB;MAAA,oD;MAAA,mB;QAO8C,iB;MAAA,C;KAP9C,C;8FASA,yB;MAAA,wE;MAAA,mB;QAQ4D,2B;MAAA,C;KAR5D,C;;MAsbID,gBAA7C,qBAAoB,YAAY,KAAM,OAAIB,CAAPB,C

;MAAqD,wB;MAArD,OUHO,S;IVGP,C;wFAEJ,yB;MAAA,4D;MAAA,mB;QAKsD,qB;MAAA,C;KALtD,C;;MA a8G,gBAAvC,eAAc,YAAY,KAAM,OAAIB,CAAd,C;MAA+C,wB;MAA/C,OUIB5D,S;IVkB4D,C;4FAEvE,yB;M AAA,wE;MAAA,mB;QAK8D,2B;MAAA,C;KAL9D,C;;MAiB+E,OAAm,QAAN,KAAM,EAAM,qBAAc,YAAY, KAAM,OAAIB,CAAd,CAAN,C;IAAN,C;;MAS3E,IAAI,eAAe,CAAnB,C;QACI,OAAO,eAAe,CAAf,I;;MAEX,IA AI,eAAe,UAAAnB,C;QACI,OAAO,gBA Ae,eAAe,CAAf,IAAf,K;;MAEX,OAAO,U;IACX,C;;;MAM+D,QAAC,mB ;IAAD,C;;MAa3D,OAAO,qBAAgB,mB;IAC3B,C;sFAEA,yB;MAAA,oD;MAAA,4B;QAM2D,uCAAQ,U;MAAR, C;KAN3D,C;;MAiBI,OAAI,mBAAJ,GAAe,cAAf,GAAmC,S;IAAnC,C;yFAEJ,yB;MAuBA,kC;MAAA,8B;MAvB A,iC;QA8BiC,Q;QAxB2E,OAwBxD,CAAnB,wDAAmB,oBAxB0E,GAwBpE,C;MAxBwD,C;KAN5G,C;+EAQA, yB;MAAA,kC;MAAA,8B;MAAA,iC;QAKiC,Q;QAA7B,OAAgD,CAAnB,wDAAmB,YAAI,GA AJ,C;MAAhD,C; KALJ,C;;MAYI,sBAAI,GA AJ,EAAS,KAAT,C;IACJ,C;4FAEA,yB;MAAA,kC;MAAA,8B;MAAA,iC;QAOiC,Q;Q AA7B,OAAgD,CAAnB,wDAAmB,oBAAY,GA AZ,C;MAAhD,C;KAPJ,C;;MAkBsG,OAAA,SAAK,qBAAc,KAA d,C;IAAL,C;kFAGtG,yB;MAAA,gD;MAAA,8B;MAAA,iC;QASiC,Q;QAA7B,OAAuD,CAA1B,+DAA0B,eAAO, GAAP,C;MAAvD,C;KATJ,C;;MA sBoE,oB;IAAA,C;;MAaA,sB;IAAA,C;kFAEpE,yB;MAAA,6B;MAAA,4B;QAI gE,qBAAK,aAAL,EA AU,eAAV,C;MAAA,C;KAJhE,C;;MAYiF,Q;MAAA,mCAAI,GA AJ,oBAAY,c;IAAZ,C;uG AGjF,yB;MAAA,gB;MAAA,8B;MAAA,+C;QAMe,Q;QALX,YAAY,oBAAI,GA AJ,C;QACZ,IAAI,iBA AiB,CAA C,4BAAY,GA AZ,CAAtB,C;UACI,OAAO,c;;;UAGP,OAAO,sE;;MAEF,C;KARA,C;;MAoBkD,uCAAqB,GAArB, C;IAAA,C;;MAYvC,Q;MADP,YAAY,oBAAI,GA AJ,C;MA CL,IAAI,aAAJ,C;QACH,aAAa,c;QACb,sBAAI,GA AJ ,EAAS,MAAT,C;QACA,a;;QAEA,Y;;MALJ,W;IAOJ,C;;MAQwF,OAAA,iBAAQ,W;IAAR,C;;MAQwB,OAAA,i BAAQ,W;IAAR,C;;MYqwB5F,Q;MAAA,OZ9vBT,iBY8vBS,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QZ9vBa ,WY+vBb,aAAGB,OZ/vBe,IY+vB/B,EZ/vBsC,SY+vBZ,CAAe,OAAf,CAA1B,C;;MZ/vBhB,OAA6B,W;IACjC,C;; MY6vBoB,Q;MAAA,OZnvBT,iBYmvBS,W;MAAhB,OAAgB,cAAhB,C;QAAgB,yB;QZnvBa,WYovBb,aZpvB0B ,SYovBtB,CAAY,OAAZ,CAAJ,EAAYC,OZpvBC,MYovB1C,C;;MZpvBhB,OAA6B,W;IACjC,C;;MAMyB,Q;MA ArB,wBAAqB,KAARb,gB;QAAqB,aAAA,KAARb,M;QAAK,IAAC,yBAAD,EAAM,2B;QACP,sBAAI,GA AJ,EA AS,KAAT,C;;IAER,C;;MAMyB,Q;MAAA,uB;MAArB,OAAqB,cAArB,C;QAAqB,wB;QAAhB,IAAC,yBAAD,E AAM,2B;QACP,sBAAI,GA AJ,EAAS,KAAT,C;;IAER,C;;MAMyB,Q;MAAA,uB;MAArB,OAAqB,cAArB,C;QA AqB,wB;QAAhB,IAAC,yBAAD,EAAM,2B;QACP,sBAAI,GA AJ,EAAS,KAAT,C;;IAER,C;wFAEA,yB;MAAA,0 D;MAAA,uE;MAAA,uC;QASW,kBAAY,mBAAoB,YAAY,cAAZ,CAApB,C;QY4sBH,Q;QAAA,OZ9vBT,iBY8v BS,W;QAAhB,OAAgB,cAAhB,C;UAAgB,yB;UZ9vBa,WY+vBb,aAAGB,OZ/vBe,IY+vB/B,EZ7sB2C,SY6sBjB,C AAe,OAAf,CAA1B,C;;QZ7sBhB,OAID6B,W;MAMdJ,C;KAVA,C;oFAYA,yB;MAAA,0D;MAAA,uE;MAAA,u C;QAYW,kBAAU,mBAAoB,YAAY,cAAZ,CAApB,C;QY6rBD,Q;QAAA,OZnvBT,iBYmvBS,W;QAAhB,OAAg B,cAAhB,C;UAAgB,yB;UZnvBa,WYovBb,aZ9rByC,SY8rBrC,CAAY,OAAZ,CAAJ,EAAYC,OZpvBC,MYovB1C ,C;;QZ9rBhB,OAtd6B,W;MAuDjC,C;KAbA,C;oFAeA,yB;MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;Q ACC,OAAA,SA3FsE,QAAQ,W;QA2F5F,OAAc,cAAAd,C;UAAc,uB;UACV,IAAI,UAAU,KAAM,IAAhB,CAAJ,C; YACI,MAAO,aAAI,KAAM,IAAV,EAAe,KAAM,MAArB,C;;QAGf,OAAO,M;MACX,C;KAdA,C;8FAgBA,yB; MAAA,wE;MAAA,uC;QAQkB,Q;QADd,aAAa,oB;QACC,OAAA,SA3GsE,QAAQ,W;QA2G5F,OAAc,cAAAd,C;U AAc,uB;UACV,IAAI,UAAU,KAAM,MAAhB,CAAJ,C;YACI,MAAO,aAAI,KAAM,IAAV,EAAe,KAAM,MAAr B,C;;QAGf,OAAO,M;MACX,C;KAdA,C;;MAwBoB,Q;MAAA,OAAA,SA3HoE,QAAQ,W;MA2H5F,OAAgB,cA AhB,C;QAAgB,yB;QACZ,IAAI,UAAU,OAAV,CAAJ,C;UACI,WAAY,aAAI,OAAQ,IAAZ,EAAiB,OAAQ,MAA zB,C;;MAGpB,OAAO,W;IACX,C;qFAEA,yB;MAAA,wE;MAAA,uC;QAOW,kBAAS,oB;QAFa,Q;QAAA,OA3 HoE,iBAAQ,W;QA2H5F,OAAgB,cAAhB,C;UAAgB,yB;UACZ,IAcmC,SAdB,CAAU,OAAV,CAAJ,C;YACI,W AAY,aAAI,OAAQ,IAAZ,EAAiB,OAAQ,MAAzB,C;;QAapB,OAVO,W;MAWX,C;KARA,C;;MAiBoB,Q;MAAA ,OAAA,SAPJoE,QAAQ,W;MAoJ5F,OAAgB,cAAhB,C;QAAgB,yB;QACZ,IAAI,CAAC,UAAU,OAAV,CAAL,C; UACI,WAAY,aAAI,OAAQ,IAAZ,EAAiB,OAAQ,MAAzB,C;;MAGpB,OAAO,W;IACX,C;2FAEA,yB;MAAA,w E;MAAA,uC;QAOW,kBAAY,oB;QAFH,Q;QAAA,OAPJoE,iBAAQ,W;QAoJ5F,OAAgB,cAAhB,C;UAAgB,yB;U ACZ,IAAI,CAckC,SAdjC,CAAU,OAAV,CAAL,C;YACI,WAAY,aAAI,OAAQ,IAAZ,EAAiB,OAAQ,MAAzB,C;; QAapB,OAVO,W;MAWX,C;KARA,C;;MAkBqB,IAAN,I;MADX,IAAI,oCAAJ,C;QACW,QAAM,cAAN,C;eAC H,C;YAAK,iB;YAAL,K;eACA,C;YAAK,aAAU,8BAAJ,GAakB,sBAAK,CAAL,CAAIB,GAA+B,oBAAW,OAAh D,C;YAAL,K;kBACQ,0BAAM,qBAAoB,YAAY,cAAZ,CAApB,CAAN,C;YAHL,K;;QAAP,W;;MAMJ,OAAoC,o

BAA7B,mBAAM,oBAAN,CAA6B,C;IACxC,C;;MAMwB,SAApB,WAAoB,Y;MAApB,kB;IAAA,C;;MAQyD,QAAM,gBAAN,C;aACzD,C;UADyD,OACpD,U;aACL,C;UAFyD,OAEPD,MAAM,UAAK,CAAL,CAAN,C;gBAFoD,OAGjD,mBAAM,qBAAoB,YAAy,gBAAZ,CAApB,CAAN,C;;IAHiD,C;;MAURc,OAAPB,WAAoB,Y;MAApB,kB;IAAA,C;;MAQwD,OAA6B,oBAA7B,mBAAM,oBAAN,CAA6B,C;IAA7B,C;;MAMPc,SAApB,WAAoB,Y;MAApB,kB;IAAA,C;;MAQiD,QAAM,cAAN,C;aACjD,C;UADiD,OAC5C,U;aACL,C;UAFiD,OaxY8B,uB;gBbwY9B,OAGzC,uB;;IAHyC,C;;MAYc,4BAAc,SAAd,C;IAAA,C;;MAO/D,WAAoB,0B;MAApB,kB;IAAA,C;;MASA,Q;MAAA,IAAI,SAAK,UAAT,C;QAAA,OAAoB,MAAM,IAAN,C;;QAAqC,kBAAPB,qBAAc,SAAd,C;QAA4B,wBAAS,UAAT,EAAqB,WAArB,C;QAAjE,OU5eO,W;;MV4eP,W;IAAA,C;;MASA,Q;MAAA,IAAI,SAAK,UAAT,C;QAAA,OAA0B,MAAN,KAAM,C;;QAAiC,kBAAPB,qBAAc,SAAd,C;QAA4B,4B;QAAne,OUrfO,W;;MVqfP,W;IAAA,C;;MASA,Q;MAAA,IAAI,SAAK,UAAT,C;QAAA,OAA0B,QAAN,KAAM,C;;QAAiC,kBAAPB,qBAAc,SAAd,C;QAA4B,0B;QAAne,OU9fO,W;;MV8fP,W;IAAA,C;;MASoB,kBAAPB,qBAAc,SAAd,C;MAA4B,4B;MAA5B,OAA4C,oBUvgBrC,WVugBqC,C;IAA5C,C;;MASoB,kBAAPB,qBAAc,SAAd,C;MAA4B,+B;MAA5B,OUhhBO,W;IVghBP,C;;MAQA,sBAAI,IAAK,MAAT,EAAgB,IAAK,OAArB,C;IACJ,C;4FAEA,yB;MAAA,gD;MAAA,mC;QAKI,kBAAO,KAAP,C;MACJ,C;KANA,C;4FAQA,yB;MAAA,gD;MAAA,mC;QAKI,kBAAO,KAAP,C;MACJ,C;KANA,C;MAAI,yBAAO,GAAP,C;IACJ,C;;MASwB,kBAAf,aAAL,SAAK,C;MASCL,6B;MatCA,OAA+C,oBUlkBxC,WVkkBwC,C;IAA/C,C;;MAUoB,kBAAf,aAAL,SAAK,C;MAqCK,YAAL,gBAAK,O;MArCV,OAAgD,oBU5kBzC,WV4kByC,C;IAAhD,C;;MAUoB,kBAAf,aAAL,SAAK,C;MAoCK,YAAL,gBAAK,O;MApCV,OAAgD,oBUtlBzC,WVslByC,C;IAAhD,C;;MAUoB,kBAAf,aAAL,SAAK,C;MAmCK,YAAL,gBAAK,O;MAncv,OAAgD,oBUhmBzC,WVgmByC,C;IAAhD,C;;MAQA,uBAAO,GAAP,C;IACJ,C;8FAEA,yB;MAAA,sD;MAAA,kC;QAMc,UAAV,SAAK,KAAK,EA AU,IAAV,C;MACd,C;KAPA,C;8FASA,yB;MAAA,sD;MAAA,kC;QAMc,UAAV,SAAK,KAAK,EAAU,IAAV,C;MACd,C;KAPA,C;8FASA,yB;MAAA,sD;MAAA,kC;QAMc,UAAV,SAAK,KAAK,EAAU,IAAV,C;MACd,C;KAPA,C;;MAWsD,QAAM,cAAN,C;aACID,C;UADkD,OAC7C,U;aACL,C;UAFkD,gB;gBAAA,OAG1C,S;;IAH0C,C;;MgJvsBID,IAAI,wCAAJ,C;QACI,OAAO,SAAK,4BAAqB,GAARb,C;MAET,4B;MhJgQI,Q;MALX,YAAy,oBgJ3Pa,GhJ2Pb,C;MACZ,IAAI,iBAAiB,CAAC,4BgJ5PG,GhJ4PH,CAAtB,C;QgJ5PgC,MAAM,2BAAuB,wCAAvB,C;;QhJgQIC,2BAAO,sE;;MgJhQX,+B;IACJ,C;;MAYQ,kBADE,SACf,kB;QADJ,OACkC,YAAT,SAAK,IAAI,EAA Y,YAAZ,C;;QADIC,OAey,uBAAmB,SAAnB,EAAyB,YAAzB,C;IAFZ,C;;MAGBI,kBADE,SACf,yB;QADJ,OA CyC,cAAT,SAAK,IAAI,EAAy,YAAZ,C;;QADzC,OAey,8BAA0B,SAA1B,EAAgC,YAAhC,C;IAFZ,C;;;;MA gBuC,wB;MAAoC,0B;IAArC,C;;MAAC,yB;IAAA,C;;MACK,OAAI,OAAJ,QAAL,EAAO,KAAP,C;IAAJ,C;;MAC b,OAAI,SAAJ,QAAL,C;IAAJ,C;;MACG,OAAA,QAAL,W;IAAJ,C;;MACH,OAAA,QAAL,K;IAAJ,C;;MACG,OAA A,QAAL,U;IAAJ,C;;MACU,OAAA,QAAL,mBAAY,GAAZ,C;IAAJ,C;;MACoB,OAAA,QAAL,qBAAc,KAAd,C;IA AJ,C;;MACjC,OAAA,QAAL,WAAI,GAAJ,C;IAAJ,C;;MACG,OAAA,QAAL,K;IAAJ,C;;MACS,OAAA,QAAL,O;I AAJ,C;;MACQ,OAAA,QAAL,Q;IAAJ,C;;MAEA,gBAAJ,Q;MAAI,4B;MhJwMxC,Q;MALX,YAAy,oBgJnMyD,G hJmMzD,C;MACZ,IAAI,iBAAiB,CAAC,4BgJpM+C,GhJoM/C,CAAtB,C;QACI,2BgJrMwE,mB;;QhJwMxE,2BA AO,sE;;MgJxMoC,+B;IAAA,C;;MAGL,wB;MAA2C,0B;IAA5C,C;;MAAC,yB;IAAA,C;;MACE,OAAI,OAAJ,Q AAI,EAAO,KAAP,C;IAAJ,C;;MACb,OAAI,SAAJ,QAAL,C;IAAJ,C;;MACG,OAAA,QAAL,W;IAAJ,C;;MACH,O AAA,QAAL,K;IAAJ,C;;MACG,OAAA,QAAL,U;IAAJ,C;;MACU,OAAA,QAAL,mBAAY,GAAZ,C;IAAJ,C;;MAC oB,OAAA,QAAL,qBAAc,KAAd,C;IAAJ,C;;MACjC,OAAA,QAAL,WAAI,GAAJ,C;IAAJ,C;;MACU,OAAA,QAAL ,K;IAAJ,C;;MACS,OAAA,QAAL,O;IAAJ,C;;MACsB,OAAA,QAAL,Q;IAAJ,C;;MAE/B,OAAA,QAAL,aAAI,GA A J,EAAS,KAAT,C;IAAJ,C;;MACP,OAAA,QAAL,cAAO,GAAP,C;IAAJ,C;;MACS,QAAL,gBAAO,IAAP,C;IAAJ,C ;MACpB,QAAL,Q;IAAJ,C;;MAE4B,gBAAJ,Q;MAAI,4B;MhJmLxC,Q;MALX,YAAy,oBgJ9KyD,GhJ8KzD,C;M ACZ,IAAI,iBAAiB,CAAC,4BgJ/K+C,GhJ+K/C,CAAtB,C;QACI,2BgJhLwE,mB;;QhJmLxE,2BAAO,sE;;MgJnLo C,+B;IAAA,C;;oFCxFnD,yB;MAAA,8D;MAAA,8B;MAAA,qC;QAUiC,Q;QAA7B,OAA2D,CAA9B,sEAA8B,eA AO,OAAP,C;MAA3D,C;KAVJ,C;wFAYA,yB;MAAA,8D;MAAA,8B;MAAA,sC;QASiC,Q;QAA7B,OAA2D,CA A9B,sEAA8B,oBAAU,QAAY,C;MAA3D,C;KATJ,C;wFAWA,yB;MAAA,8D;MAAA,8B;MAAA,sC;QASiC,Q;Q AA7B,OAA2D,CAA9B,sEAA8B,oBAAU,QAAY,C;MAA3D,C;KATJ,C;;MAiB6D,kCAAS,KAAT,C;IAAA,C;;M AOzD,SAAK,WAAI,OAAJ,C;IACT,C;4FAEA,yB;MAAA,gD;MAAA,sC;QAKS,OAAAL,SAAK,EAAO,QAAP,C; MACT,C;KANA,C;4FAQA,yB;MAAA,gD;MAAA,sC;QAKS,OAAAL,SAAK,EAAO,QAAP,C;MACT,C;KANA,C;

4FAQA,yB;MAAA,gD;MAAA,sC;QAKS,OAAL,SAAK,EAAO,QAAP,C;MACT,C;KANA,C;;MAaI,SAAK,cAA
O,OAAP,C;IACT,C;8FAEA,yB;MAAA,sD;MAAA,sC;QAKS,UAAL,SAAK,EAAU,QAAP,C;MACT,C;KANA,C
;8FAQA,yB;MAAA,sD;MAAA,sC;QAKS,UAAL,SAAK,EAAU,QAAP,C;MACT,C;KANA,C;8FAQA,yB;MAAA
,sD;MAAA,sC;QAKS,UAAL,SAAK,EAAU,QAAP,C;MACT,C;KANA,C;;MAYU,IAIe,I;MAHjB,kBADE,QACF,
c;QAAiB,OOAO,yBAAO,QAAP,C;;QAEpB,aAAsB,K;QACT,0B;QAAb,OAAa,cAAb,C;UAAa,sB;UACT,IAAI,o
BAAI,IAAJ,CAAJ,C;YAAe,SAAS,I;;QAC5B,OOAO,M;;IAGnB,C;;MAOiB,Q;MADb,aAAsB,K;MACT,0B;MAA
b,OAAa,cAAb,C;QAAa,sB;QACT,IAAI,oBAAI,IAAJ,CAAJ,C;UAAe,SAAS,I;;MAE5B,OOAO,M;IACX,C;;MA
MI,OOAO,yBAAGB,OAAT,QAAS,CAAhB,C;IACX,C;;MAKkF,gCAAc,SAAd,EAAyB,IAAzB,C;IAAA,C;;MAK
A,gCAAc,SAAd,EAAyB,KAAzB,C;IAAA,C;;MAG9E,iBAAa,KAAb,C;MvI7FgB,kBuI8FX,oB;MACD,OOAO,q
BAAP,C;QACI,IAAI,UAAU,kBAAV,6BAAJ,C;UACI,oB;UACA,WAAS,I;;MAGrB,OOAO,Q;IACX,C;;MAK8E,
kCAAc,SAAd,EAAyB,IAAzB,C;IAAA,C;;MAKA,kCAAc,SAAd,EAAyB,KAAzB,C;IAAA,C;;MAI9D,UAGS,M
AHT,EAcY,MADz,EAc6B,M;MAfzC,IAAI,uCAAJ,C;QACI,OOAO,cAA5B,sEAA4B,EAAc,SAAd,EAAyB,uBA
AzB,C;MAExC,iBAAsB,C;MACD,oC;MAArB,qBAAkB,CAAIb,mC;QACI,cAAc,sBAAK,SAAL,C;QACd,IAAI,
UAAU,OOAV,MAAsB,uBAA1B,C;UACI,Q;QAEJ,IAAI,eAAc,SAAIb,C;UACI,sBAAK,UAAL,EAAMb,OOAnB
,C;QAEJ,+B;;MAEJ,IAAI,aAAa,cAAjB,C;QACwB,oC;QAAiB,mB;QAArC,oE;UACI,2BAAS,WAAT,C;QAEJ,O
AAO,I;;QAEp,OOAO,K;;IAEf,C;;MAMW,iBAAMb,gCAAT,QAAS,EAAgC,SAAhC,C;MAnLG,Q;MAmL7B,O
AnL2D,CAA9B,sEAA8B,oBAAU,UAAV,C;IAoL/D,C;;MAMI,UAAmB,YAAT,QAAS,C;MACnB,OIJ7CwD,CkJ
6CjD,GIJ7CkD,UkJ6CID,IAAoB,4BAAU,GAAV,C;IAC/B,C;;MAMI,OxJgtMO,EwJhtMA,QxJwoMA,YAAQ,CA
wER,CwJhtMA,IAAyB,4BAAMb,UAAT,QAAS,CAAnB,C;IACpC,C;;MAMW,iBAAMb,gCAAT,QAAS,EAAgC,
SAAhC,C;MA9LG,Q;MA8L7B,OA9L2D,CAA9B,sEAA8B,oBAAU,UAAV,C;IA+L/D,C;;MAMI,IxJksMO,EwJls
MH,QxJ0nMG,YAAQ,CAwER,CwJlsMP,C;QACI,OOAO,4BAAMb,UAAT,QAAS,CAAnB,C;;QAEp,OOAO,wB;
IACf,C;;MAMI,UAAmB,YAAT,QAAS,C;MACnB,IJ7EwD,CkJ6EpD,GIJ7EqD,UkJ6EzD,C;QACI,OOAO,4BAA
U,GAAV,C;;QAEp,OOAO,wB;IACf,C;;MAGI,alJpFwD,CAAC,mB;MkJqFzD,iB;MACA,OOAO,M;IACX,C;;MA
SI,aAAU,2BAAV,OOA2B,CAA3B,M;QACI,QAAQ,MAAO,iBAAQ,IAAI,CAAJ,IAAR,C;QACf,WAAW,sBAAK
,CAAL,C;QACX,sBAAK,CAAL,EAAU,sBAAK,CAAL,CAAV,C;QACA,sBAAK,CAAL,EAAU,IAAV,C;;IAER,
C;;MAO+E,kBAAhB,0B;MAAwB,8B;MAAxB,OvIzMpD,W;IuIyMoD,C;;MCIRiB,uB;MAAjC,0B;IAAD,C;;MA
CX,OOAA,eAAS,K;IAAT,C;;MACG,mCAAS,0BAAoB,KAApB,CAAT,C;IAAA,C;;MAGgC,8B;MAAxC,0B;IA
AD,C;;MACM,OOAA,eAAS,K;IAAT,C;;MACG,mCAAS,0BAAoB,KAApB,CAAT,C;IAAA,C;;MAEX,eAAS,Q;I
AAT,C;;MACgB,OOAA,eAAS,kBAAS,0BAAoB,KAApB,CAAT,C;IAAT,C;;MAEO,OOAA,eAAS,aAAI,0BAAo
B,KAApB,CAAJ,EAAgC,OOAhC,C;IAAT,C;;MAE1C,eAAS,aAAI,2BAAqB,KAArB,CAAJ,EAAiC,OOAjC,C;IA
Cb,C;;MAIgB,Q;MAAA,kC;MAAhB,IAAa,CAAT,0BAAJ,C;QAAA,OOA2B,8BAAy,KAAZ,I;;QAAuB,MAAM,
8BAA0B,mBAAGB,KAAhB,2BAA0C,gBAAG,2BAAH,CAA1C,OOA1B,C;IAAxD,C;;MAGgB,Q;MAAA,qB;MA
AhB,IAAa,CAAT,0BAAJ,C;QAAA,OOAsB,iBAAO,KAAp,I;;QAAkB,MAAM,8BAA0B,oBAAiB,KAAjB,2BAA
2C,gBAAG,cAAH,CAA3C,OOA1B,C;IAA9C,C;;MAQ2C,gCAAqB,SAArB,C;IAAA,C;;MAQc,wBAAa,SAAb,C;
IAAA,C;;M7I9B0B,wC;IARvF,C;IAAA,0D;MAS2C,8B;IAAA,C;IAT3C,gF;;M8Ic8G,mB;QAAE,kBAAS,aAAT
,C;MAAF,C;IAAA,C;;MAAT,oCAAS,sBAAT,C;IAAA,C;8FAErG,yB;MAAA,kD;M9IhBA,kC;MAAA,0C;;QAQ
uF,wC;MARvF,C;MAAA,wD;QAS2C,8B;MAAA,C;MAT3C,8E;;Q8ImB2I,mB;UAAE,gBAAS,qBAAT,C;QAAF,
C;MAAA,C;MAH3I,gC;QAGkI,kCAAS,mCAAT,C;MAAA,C;KAHII,C;;MAaI,eAAe,6B;MACf,oBAA0B,+BAA
N,KAAAM,EAAwC,QAAXC,EAA+D,QAA/D,C;MAC1B,OOAO,Q;IACX,C;8FAEA,yB;MAAA,kD;MAAA,gC;QA
GkI,gBAAS,aAAT,C;MAAA,C;KAHII,C;;IAGB0C,C;;MA2BIC,IAAI,uCAA0B,QAAS,UAAvC,C;QAAkD,M;MA
CID,OOAO,sBAAS,QAAS,WAAIB,e;IACX,C;;MAUqD,6BAAS,QAAS,WAAIB,e;IAAA,C;;MAed,wB;MAC
vC,eAAoB,C;MACpB,mBAA4B,I;MAC5B,sBAAyC,I;MACzC,gBAAoC,I;IAJxC,C;;MAOQ,OOAO,IAAP,C;QA
CI,QAAM,YAAN,C;eACI,C;YAAA,K;eACA,C;YACI,IAAI,kCAAe,UAAAnB,C;cACI,eAAQ,C;cACR,OOAO,I;;c
AEP,sBAAe,I;;YALvB,K;eAOA,C;YAAc,OOAO,K;eACrB,C;eAAA,C;YAAgC,OOAO,I;kBAC/B,MAAM,yB;;Q
AGIB,eAAQ,C;QACR,WAAW,4B;QACX,gBAAW,I;QACX,I/FhGR,oBD6CQ,WgGmDY,kBhGnDZ,CC7CR,C;;I
+FkGA,C;;MAGU,IASe,I;MATrB,QAAM,YAAN,C;aACI,C;aAAA,C;UAAcC,OOAO,qB;aAC7C,C;UACI,eAAQ,
C;UACR,OOAO,kCAAe,O;aAE1B,C;UACI,eAAQ,C;UACR,aACa,mF;UACb,mBAAy,I;UACZ,OOAO,M;gBAE
H,MAAM,yB;;IAEtB,C;;MAGI,IAAI,CAAC,cAAL,C;QAAGB,MAAM,6B;;QAA8B,OOAO,W;IAC/D,C;;MAE4C

,QAAM,YAAN,C;aACxC,C;UADwC,OAC1B,6B;aACd,C;UAFwC,OAExB,6BAAsB,sBAAtB,C;gBAFwB,OAGh
C,6BAAsB,uCAAOc,YAA1D,C;;IAHgC,C;;MAUK,oB;QACzC,wCAAW,C;QAAX,OACA,yB;MAFyC,C;IAAA,
C;;MAF7C,mBAAY,K;MACZ,eAAQ,C;MACR,OAA6C,0CAAtC,c;IAIX,C;;MAMiD,oB;QACzC,wCAAW,C;QA
AX,OACA,yB;MAFyC,C;IAAA,C;;MAH7C,IAAI,CAAC,QAAS,UAAAd,C;QAAyB,M;MACzB,sBAAE,Q;MACf,e
AAQ,C;MACR,OAA6C,6CAAtC,c;IAIX,C;;MhGrBO,Q;MADP,egG0BI,MhG1BJ,C;MACO,QgGyBH,MhGzBG,
+D;MgG0BH,eAAQ,C;IACZ,C;;MAGY,0C;IAAA,C;;M9IpLuE,wC;IARvF,C;IAAA,0D;MAS2C,8B;IAAA,C;IA
T3C,gF;sFAAA,yB;MAAA,kC;MAAA,0C;;QAQuF,wC;MARvF,C;MAAA,wD;QAS2C,8B;MAAA,C;MAT3C,8E;
MAAA,2B;QAQuF,2C;MAAA,C;KARvF,C;;MAiBgE,mB;QAAE,sB;MAAF,C;IAAA,C;;MAAT,OAAkB,2CAAT
,+BAAS,E;IAAI,B,C;;MAOM,OAAI,QZy0MtD,YAAQ,CYz0M0C,GAAwB,eAAxB,GAAAD,WAAT,QAAS,C;IA
AtD,C;;MAKf,kC;IAAA,C;;MAE9C,6B;IAAA,C;;MACiD,kC;IAAA,C;;MACjB,kC;IAAA,C;;MACA,kC;IAAA,C
;;;MAHhC,oC;QAAA,mB;;MAAA,6B;IAAA,C;oFAMA,yB;MAAA,2D;MAAA,4B;QAM4D,uCAAQ,e;MAAR,C
;KAN5D,C;;MAGb4F,wC;MAAA,6B;MAAA,yB;MAAA,wC;MAAA,wD;MAAA,kC;IAO5F,C;;;;cANI,eAAe,
uBAAa,W;cAC5B,IAAI,QAAS,UAAb,C;gBACI,gB;gCAAA,sCAAS,QAAT,O;oBAAA,kBcjER,yB;yBdiEQ,yB;g
BAAA,Q;;gBAEA,gB;gCAAA,sCAAS,iCAAT,O;oBAAA,kBcnER,yB;yBdmEQ,yB;gBAAA,Q;;;;cAJJ,W;;cAA
A,W;;;;;;;;;;IAMJ,C;;MAP4F,yD;uBAAA,uG;YAAA,S;iBAAA,Q;;iBAAA,uB;MAAA,C;IAAA,C;;MAAT,gB
AAS,uCAAT,C;IAAA,C;;MAcX,OAAA,EAAG,W;IAAW,C;;MAAxB,4BAAQ,cAAR,C;IAAA,C;;MAQU,OAAA,
EAAG,W;IAAW,C;;MAAxB,4BAAQ,gBAAR,C;IAAA,C;;MAMxB,S;IAAG,C;;MAFzB,Q;MADZ,IAAI,8CAAJ,
C;QACI,OAA4C,CAApC,2EAAoC,kBAAQ,QAAR,C;;MAEhD,OAAO,uBAAmB,SAAAnB,EAAyB,gBAAZB,EAA
iC,QAAjC,C;IACX,C;;MAYiB,Q;MAFb,YAAY,gB;MACZ,YAAY,gB;MACC,2B;MAAb,OAAa,cAAb,C;QAAa,s
B;QACT,KAAM,WAAL,IAAK,MAAT,C;QACN,KAAM,WAAL,IAAK,OAAT,C;;MAEV,OAAO,UAAAS,KAAT,C;
IACX,C;;MAWI,wB;QAAA,WAAGC,I;MADhC,0B;MACA,0B;MACA,4B;IAH+B,C;;MAMQ,oD;MACnC,gBAA
e,iCAAS,W;MACxB,iBAAqB,E;MACrB,gBAAmB,I;IAHgB,C;;MAM/B,OAAO,aAAS,UAAhB,C;QACI,WAAW,
aAAS,O;QACpB,IAAI,wCAAU,IAAV,MAAmB,sCAAvB,C;UACI,gBAAW,I;UACX,iBAAy,C;UACZ,M;;MAG
R,iBAAy,C;IACb,C;;MAWW,Q;MARP,IAAI,mBAAa,EAajB,C;QACI,iB;MACJ,IAAI,mBAAa,CAAjB,C;QAC
I,MAAM,6B;MACV,aAAa,a;MACb,gBAAW,I;MACX,iBAAy,E;MAEZ,OAAO,yE;IACX,C;;MAGI,IAAI,mBAA
a,EAajB,C;QACI,iB;MACJ,OAAO,mBAAa,C;IACxB,C;;MAjCmC,yD;IAAA,C;;MA2C/B,0B;MAAmC,gC;IAA
/C,C;;MAC2C,0D;MACnC,gBAAE,oCAAS,W;IADW,C;;MAG/B,OAAO,6CAAY,aAAS,OAARb,C;IACX,C;;MA
GI,OAAO,aAAS,U;IACpB,C;;MARmC,4D;IAAA,C;;MAYnC,OAAO,uBAA4B,eAA5B,EAAsC,kBAATC,EAAM
D,QAAAnD,C;IACX,C;;MASQ,0B;MAAmC,gC;IAA/C,C;;MAC2C,wE;MACnC,gBAAE,2CAAS,W;MACxB,aAA
Y,C;IAFuB,C;;MAIO,Q;MAATC,OAAO,oDAAY,oBAAmB,iBAAnB,EAAMb,yBAAnB,QAAZ,EAAyC,aAAS,O
AAID,C;IACX,C;;MAGI,OAAO,aAAS,U;IACpB,C;;MATmC,mE;IAAA,C;;MAkB/B,0B;IAAZ,C;;MAEQ,gBA
Ae,gCAAS,W;MACxB,aAAY,C;IAFqC,C;;MAIN,Q;MAAvC,OAAO,iBAAa,oBAAmB,iBAAnB,EAAMb,yBAAn
B,QAAb,EAA0C,aAAS,OAAnD,C;IACX,C;;MAGI,OAAO,aAAS,U;IACpB,C;;MATiD,wD;IAAA,C;;MAoBrD,
4B;MACA,4B;MACA,4B;IAHJ,C;;MAK2C,gD;MACnC,iBAAgB,gCAAU,W;MAC1B,iBAAgB,gCAAU,W;IAFS,
C;;MAI/B,OAAO,sCAAU,cAAU,OAAPb,EAA4B,cAAU,OAATC,C;IACX,C;;MAGI,OAAO,cAAU,UAAV,IAAu
B,cAAU,U;IAC5C,C;;MATmC,uD;IAAA,C;;MAEvC,0B;MACA,gC;MACA,0B;IAHJ,C;;MAK2C,sD;MACnC,g
BAAE,kCAAS,W;MACxB,oBAAiC,I;IAFE,C;;MAK/B,IAAI,CAAC,2BAAL,C;QACI,MAAM,6B;MACV,OAAO,
gCAAE,O;IAC1B,C;;MAGI,OAAO,2B;IACX,C;;MAGQ,Q;MAAJ,IAAI,iEAA2B,KAA/B,C;QACI,oBAAe,I;MAE
nB,OAAO,yBAAP,C;QACI,IAAI,CAAC,aAAS,UAAAd,C;UACI,OAAO,K;;UAEP,cAAc,aAAS,O;UACvB,uBAAU
B,wCAAS,2CAAY,OAAZ,CAAT,C;UACvB,IAAI,gBAAiB,UAArB,C;YACI,oBAAe,gB;YACf,OAAO,I;;MAIn
B,OAAO,I;IACX,C;;MA/BmC,0D;IAAA,C;;;;MAGDvC,0B;MACA,8B;MACA,0B;MC5QA,IAAI,EDgRQ,qBA
Ac,CChRtB,CAAJ,C;QACI,cD+Q2B,+CAA4C,iB;QC9QvE,MAAM,gCAAYB,OAAQ,WAAjC,C;;MAFV,IAAI,E
DiRQ,mBAAY,CCjRpB,CAAJ,C;QACI,gBDgRyB,6CAA0C,e;QC/QnE,MAAM,gCAAYB,SAAQ,WAAjC,C;;MA
FV,IAAI,EDkRQ,mBAAY,iBCIRpB,CAAJ,C;QACI,gBDiRkC,0DAAuD,eAAvD,WAAMe,iB;QChRrG,MAAM,g
CAAYB,SAAQ,WAAjC,C;;IDuQe,C;;MAYM,yBAAW,iBAAX,I;IAAA,C;;MAEU,OAAI,KAAK,YAAT,GAAgB,
eAAhB,GAAqC,gBAAy,eAAZ,EAAsB,oBAAa,CAAb,IAATB,EAAsC,eAATC,C;IAArC,C;;MACA,OAAI,KAAK,
YAAT,GAAgB,IAAhB,GAA0B,gBAAy,eAAZ,EAAsB,iBAATB,EAAC,oBAAa,CAAb,IAAIC,C;IAAI,B,C;;MA
Ef,wC;MAEtB,gBAAE,2BAAS,W;MACxB,gBAAE,C;IAHO,C;;MAOIB,OAAO,gBAAW,kCAAX,IAAyB,aAAS,

UAAzC,C;QACI,aAAS,O;QACT,qC;;IAER,C;;MAGI,a;MACA,OAAQ,gBAAW,gCAAZ,IAAyB,aAAS,U;IAC7C,C;;MAGI,a;MACA,IAAI,iBAAY,gCAAhB,C;QACI,MAAM,6B;MACV,qC;MACA,OAAO,aAAS,O;IACpB,C;;MAxBSb,mD;IAAA,C;;MAiC1B,0B;MACA,oB;MC5TA,IAAI,EDgUQ,gBAAS,CChUjB,CAAJ,C;QACI,cD+TsB,yCAAsC,YAAtC,M;QC9TtB,MAAM,gCAAyB,OAAQ,WAAjC,C;;IDwTgB,C;;MASE,oAAI,KAAK,YAAT,GAAGB,eAAhB,GAAqC,gBAAY,eAAZ,EAAAsB,CAAtB,EAAyB,YAAzB,C;IAArC,C;;MACA,OAAI,KAAK,YAAT,GAAGb,IAAhB,GAA0B,iBAAa,eAAb,EAAuB,CAAvB,C;IAAI1B,C;;MAGrC,YAAW,yB;MACX,gBA Ae,4BAAS,W;IAFW,C;;MAK/B,IAAI,cAAQ,CAAZ,C;QACI,MAAM,6B;MACV,6B;MACA,OAAO,aAAS,O;IACpB,C;;MAGI,OAAO,YAAO,CAAP,IAAY,aAAS,U;IACHc,C;;MABmC,oD;IAAA,C;;MAuBvC,0B;MACA,4B;IAFJ,C;;MAI2C,oD;MACnC,gBA Ae,iCAAS,W;MACxB,iBAAqB,E;MACrB,gBAAmB,I;IAHgB,C;;MAM/B,IAAI,aAAS,UAAb,C;QACI,WAAW,aAAS,O;QACpB,IAAI,wCAAU,IAAV,CAAJ,C;UACI,iBAAY,C;UACZ,gBAAW,I;UACX,M;;MAGR,iBAAY,C;IACHb,C;;MAQiB,Q;MALb,IAAI,mBAAa,EAAjB,C;QACI,iB;MACJ,IAAI,mBAAa,CAAjB,C;QACI,MAAM,6B;MACV,aACa,gF;MAGb,gBAAW,I;MACX,iBAAY,E;MACZ,OAAO,M;IACX,C;;MAGI,IAAI,mBAAa,EAAjB,C;QACI,iB;MACJ,OAAO,mBAAa,C;IACxB,C;;MANCmC,yD;IAAA,C;;MA4CvC,0B;MACA,oB;MC7YA,IAAI,EDgZQ,gBAAS,CChZjB,CAAJ,C;QACI,cD+YsB,yCAAsC,YAAtC,M;QC9YtB,MAAM,gCAAyB,OAAQ,WAAjC,C;;IDyYgB,C;;MK/TnB,SLuUmC,eAAQ,CAAR,I;MAAD,OAA4B,KAAK,CAAT,GAAY,yBAAZ,GAAuC,iBAAa,eAAb,EAAuB,EAAvB,C;IAA/D,C;;MKvUIC,SLwUmC,eAAQ,CAAR,I;MAAD,OAA4B,KAAK,CAAT,GAAY,yBAAZ,GAAuC,gBAAY,eAAZ,EAAAsB,YAAtB,EAA6B,EAA7B,C;IAA/D,C;;MAGrC,gBA Ae,4BAAS,W;MACxB,YAAW,yB;IAFwB,C;;MAM/B,OAAO,YAAO,CAAP,IAAY,aAAS,UAA5B,C;QACI,aAAS,O;QACT,6B;;IAER,C;;MAGI,a;MACA,OAAO,aAAS,O;IACpB,C;;MAGI,a;MACA,OAAO,aAAS,U;IACpB,C;;MAPBmC,oD;IAAA,C;;MA8BvC,0B;MACA,4B;IAFJ,C;;MAK2C,oD;MACnC,gBA Ae,iCAAS,W;MACxB,iBAAqB,E;MACrB,gBAAmB,I;IAHgB,C;;MAM/B,OAAO,aAAS,UAAhB,C;QACI,WAAW,aAAS,O;QACpB,IAAI,CAAC,wCAAU,IAAV,CAAL,C;UACI,gBAAW,I;UACX,iBAAY,C;UACZ,M;;MAGR,iBAAY,C;IACHb,C;;MAQqB,Q;MALjB,IAAI,mBAAa,EAAjB,C;QACI,a;MAEJ,IAAI,mBAAa,CAAjB,C;QACI,aACa,gF;QACb,gBAAW,I;QACX,iBAAY,C;QACZ,OAAO,M;;MAEX,OAAO,aAAS,O;IACpB,C;;MAGI,IAAI,mBAAa,EAAjB,C;QACI,a;MACJ,OAAO,mBAAa,CAAb,IAAkB,aAAS,U;IACtC,C;;MANCmC,yD;IAAA,C;;MAuCL,sB;MAAiC,gC;IAAIC,C;;MACM,4BAAiB,aAAO,WAAxB,EAAoC,kBAAPC,C;IAAA,C;;MAGgE,2B;MAAtE,sB;MAAiC,gC;MACIE,kBAAuB,c;IADS,C;;MAI5B,OAAO,aAAO,UAAAd,C;QACI,WAAW,aAAO,O;QACIB,UAAU,mBAAY,IAAZ,C;QAEV,IAAI,eAAS,WAAI,GAAJ,CAAb,C;UACI,mBAAQ,IAAR,C;UACA,M;;MAIR,W;IACJ,C;;MAIqC,wC;MAAuC,kC;IAAxC,C;;MACG,oD;MACnC,gBAAmB,I;MACnB,iBAAqB,E;IAFc,C;;MAK/B,gBA Ae,mBAAa,EAAjB,GAAqB,+CAArB,GAA4C,2CAAa,4BAAb,C;MACvD,iBAAgB,qBAAJ,GAAsB,CAAtB,GAA6B,C;IAC7C,C;;MAQiB,Q;MALb,IAAI,iBAAY,CAAhB,C;QACI,iB;MAEJ,IAAI,mBAAa,CAAjB,C;QACI,MAAM,6B;MACV,aAAa,8D;MAEb,iBAAY,E;MACZ,OAAO,M;IACX,C;;MAGI,IAAI,iBAAY,CAAhB,C;QACI,iB;MACJ,OAAO,mBAAa,C;IACxB,C;;MAzBmC,yD;IAAA,C;;MAwCvC,OAAW,iDAAJ,GAAwC,SAAxC,GAakD,4BAAwB,SAAxB,C;IAC7D,C;;Mac2C,qB;QAAE,6B;MAAF,C;IAAA,C;;MAAvC,OAA2D,cAAPd,sBAAkB,YAAIB,EAAgC,qCAAhC,CAAOd,C;IAC/D,C;;MAoB0B,mB;QAAE,mB;MAAF,C;IAAA,C;;MAHtB,OAAI,YAAJ,GACI,2BADJ,GAGI,sBAAkB,+BAAIB,EAA4B,YAA5B,C;IAHJ,C;;MAMBA,6BAAkB,YAAIB,EAAgC,YAAhC,C;IAAA,C;;MJ3mBJ,wB;MACI,8C;IADJ,C;;MAGgD,oCAAmB,KAAM,U;IAAzB,C;;MACb,Q;IAAA,C;;MACG,W;IAAA,C;;MAEH,Q;IAAA,C;;MACG,W;IAAA,C;;MACiB,Y;IAAA,C;;MACgB,OAAA,QAAS,U;IAAT,C;;MAEtB,kC;IAAA,C;;MAEZ,6B;IAAAA,C;;MAdrC,+B;QAAA,c;;MAAA,wB;IAAA,C;;MASBoC,6B;IAAA,C;;MAQe,OAAI,QAAS,OAAT,GAAGB,CAAPB,GAAgC,MAAT,QAAS,CAAhC,GAA6C,U;IAA7C,C;IFAEnd,yB;MAAA,mD;MAAA,mB;QAKwC,iB;MAAA,C;KALx,C;6FAOA,yB;MAAA,uE;MAAA,mB;QAQsD,2B;MAAA,C;KARtD,C;;MAeiE,OAAS,aAAT,QAAS,EAAa,qBAAC,YAAY,QAAS,OAAR,B,CAAd,CAAb,C;IAAT,C;uFAEjE,yB;MAAA,2D;MAAA,mB;QAGgD,qB;MAAA,C;KAHhD,C;;MAM2D,OAAS,aAAT,QAAS,EAAa,eAAQ,YAAY,QAAS,OAAR,B,CAAR,CAAb,C;IAAT,C;2FAE3D,yB;MAAA,uE;MAAA,mB;QAGwD,2B;MAAA,C;KAHxD,C;;MASmE,OAAS,aAAT,QAAS,EAAa,qBAAC,YAAY,QAAS,OAAR,B,CAAd,CAAb,C;IAAT,C;sFAEnE,yB;MAAA,mD;MAAA,4B;QAEkD,uCAAQ,U;MAAR,C;KAFID,C;;MAIgD,QAAM,cAAN,C;aAC5C,C;UAD4C,OACvC,U;aACL,C;UAF4C,OAEvC,MAAM,oBAAW,OAAjB,C;gBAFuC,OAGpC,S;;IAHoC,C;;MIxEuC,wC;IARvF,C;IAAA,0D;MAS2C,8B;IAAA,C;IAT3C,gF;;MCyBI,IAAI,E8I3BI,OAAO,CAAP,IAAY,OAAO,C9I2BvB,CAAJ,C;QACI,c8I3BI,aAAJ,GACI,yEADJ,GAGI,

8C;Q9IyBJ,MAAM,gCAAyB,OAAQ,WAAjC,C;;I8IvBd,C;;MAIoB,mB;QAAE,wBAAiB,gCAAjB,EAA6B,YAA7
B,EAAmC,YAAnc,EAAYC,sBAAzC,EAAYD,mBAAZD,C;MAAF,C;IAAA,C;;MADhB,oBAAoB,IAApB,EAA0B
,IAA1B,C;MACA,oCAAgB,6EAAhB,C;IACJ,C;;MAI6B,wC;MAAA,6B;MAAA,yB;MAAA,wC;MAAA,wC;MA
AA,gD;MAAA,sD;MAAA,4D;MAAA,wB;MAAA,0B;MAAA,uB;MAAA,0B;MAAA,wB;MAAA,qB;MAAA,4B;
MAAA,kC;IAkCzB,C;,,,,,;+BAjCc,0BAAO,uBAAP,I;cACV,IAAI,kBAAO,CAAX,C;oCACiB,iBAAa,uBAAb,C;
kCACF,C;gBACD,6C;gBAAV,iB;,,,;sCAaa,eAAc,uBAAd,C;gBACH,+C;gBAAV,gB;,,,,,;cAAA,KAAU,2BAAV,C
;gBAAA,gB;,,,;cAAU,kC;cACN,mBAAO,WAAI,GA AJ,C;cACP,IAAI,mBAAO,SAAX,C;gBACI,gB;gCAAA,iCA
AU,8BAAJ,GAAiB,mBAAjB,GAA6B,iBAAU,mBAAV,CAAnC,O;oBAAA,kBjI5CpB,yB;yBiI4CoB,yB;gBAAA,
Q;,,;gBADJ,gB;,,,;cAEI,mBAAO,qBAAy,uBAAZ,C;cAFX,gB;,,;cAFJ,gB;,,;cAOA,IAAI,iCAAJ,C;gBACI,gB;,,;gB
ADJ,iB;,,,;cACI,IAAO,mBAAO,KAA d,IAAqB,uBAArB,C;gBAAA,gB;,,,;cACI,gB;8BAAA,iCAAU,8BAAJ,GAAi
B,mBAAjB,GAA6B,iBAAU,mBAAV,CAAnC,O;kBAAA,kBjIIdpB,yB;uBiIkDoB,yB;cAAA,Q;,,;cACA,mBAAO,q
BAAy,uBAAZ,C;cAFX,gB;,,;cAIA,IrJyH4C,CqJzHxC,mBrJyHyC,UqJzH7C,C;gBAAyB,gB;gCAAA,iCAAM,mB
AAN,O;oBAAA,kBjIrDzC,yB;yBiIqDyC,yB;gBAAA,Q;,,;gBAAzB,iB;,,,;cA9BR,W;,,;cAyBI,iB;,,;cAxBJ,iB;,,;cAGI,
KAAU,yBAAV,C;gBAAA,iB;,,,;6BAAU,sB;cACN,IAAI,kBAAO,CAAX,C;gBAAgB,oCAAQ,CAAR,I;gBAAW,i
B;,,,;gBAA3B,iB;,,,;cACA,iBAAO,WAAI,YAAJ,C;cACP,IAAI,iBAAO,KAAP,KAAe,uBAAnB,C;gBACI,iB;gCA
AA,iCAAM,iBAAN,O;oBAAA,kBjI/BpB,yB;yBiI+BoB,yB;gBAAA,Q;,,;gBADJ,iB;,,,;cAEI,IAAI,8BAAJ,C;gBA
AiB,iBAAO,Q;gBAAa,oBAAS,iBAAU,uBAAV,C;cAC9C,kBAAO,c;cAHX,iB;,,;cAHJ,iB;,,;cASA,IrJ0IgD,CqJ1I5
C,iBrJ0I6C,UqJ1IjD,C;gBACI,IAAI,qCAAkB,iBAAO,KAAP,KAAe,uBAArC,C;kBAA2C,iB;kCAAA,iCAAM,iB
AAN,O;sBAAA,kBjIrC3D,yB;2BiIqC2D,yB;kBAAA,Q;,,;kBAA3C,iB;,,,;gBADJ,iB;,,,;cAbJ,W;,,;cAaI,iB;,,;cAZJ,iB
,,;cAgCJ,W;,,,,,;,,,,,;IAAA,C;;MAICyB,yD;uBAAA,6K;YAAA,S;iBAAA,Q;,,;iBAAA,uB;MAAA,C;IAAA,C;;M
ADzB,IAAI,CAAC,QAAS,UAA d,C;QAAYB,OAAO,2B;MACHC,OAAO,WAAkB,0EAAIB,C;IAmCX,C;;MAEiE,
uB;MAA7B,kB;MACHC,mBAA6B,C;MAC7B,eAAyB,C;IAFM,C;;MAK3B,+DAAKB,SAaIB,EAA6B,OAA7B,E
AAsC,WAAK,KAA3C,C;MACA,mBAAiB,S;MACjB,eAAa,UAAU,SAAV,I;IACjB,C;;MAGI,+DAAKB,KAAIB,E
AAyB,YAAzB,C;MAEA,OAAO,wBAAK,mBAAy,KAAZ,IAAL,C;IACX,C;;MAE+B,mB;IAAA,C;;MASc,uB;M
AArB,wB;M9IIdxB,IAAI,E8IoDQ,iBAAy,C9IpDpB,CAAJ,C;QACI,c8ImDyB,2DAAwD,a;Q9IIdjF,MAAM,gCA
AyB,OAAQ,WAAjC,C;;M8IqDV,gBAAqB,gBAAmB,aAnB,O;MACrB,oBAA8B,C;MAE9B,sBAAyB,C;IARF,
C;;MAQvB,0B;IAAA,C;MAAA,0B;IAAA,C;;MAMW,Q;MAFP,+DAAKB,KAAIB,EAAyB,SAAzB,C;MAEA,OA
AO,sBAyFmC,CAzF5B,iBAyF6B,GAzFV,KAYFU,IAAD,IAAa,aAzFhD,4D;IACX,C;;MAEe,qBAAQ,a;IAAR,C;;
MAEwB,sC;MAAS,2B;MAC5C,eAAoB,oB;MACpB,eAAoB,4B;IAFe,C;;MASnB,Q;MAJZ,IAAI,iBAAS,CAAb,C
;QACI,W;,,;QAGA,mBAAQ,sCAAQ,YAAP,4DAAR,C;QACA,eA0EkC,CA1E1B,YA0E2B,GA1Eb,CA0Ea,IAAD,
IAAa,6B;QAZe/C,mC;;IAER,C;;MAbmC,kD;IAAA,C;;MAmB0B,UAQ1C,MAR0C,EAe1C,MAf0C,EAqBtD,M;
MAtBP,aACQ,KAAM,OAAN,GAAa,IAAK,KAAtB,GAakC,UAAN,KAAM,EAAO,IAAK,KAAZ,CAAIC,GAAy
D,kD;MAE7D,WAAW,IAAK,K;MAEhB,WAAW,C;MACX,UAAU,iB;MAEV,OAAO,OAAO,IAAP,IAAe,MAA
M,aAA5B,C;QACI,OAAO,IAAP,IAAe,wBAAO,GAAP,gE;QACf,mB;QACA,iB;,,;MAGJ,MAAM,C;MACN,OAA
O,OAAO,IAAd,C;QACI,OAAO,IAAP,IAAe,wBAAO,GAAP,gE;QACf,mB;QACA,iB;,,;MAEJ,IAAI,MAAO,OAA
P,GAAC,IAAK,KAAvB,C;QAA6B,OAAO,IAAK,KAAZ,IAAoB,I;MAEjD,OAAO,uD;IACX,C;;MAGI,OAAO,qB
AAQ,gBAAa,SAAb,OAAR,C;IACX,C;;MAMI,IAAI,aAAJ,C;QACI,MAAM,6BAAsB,qBAAtB,C;;MAGV,cA6B0
C,CA7BnC,iBA6BoC,GA7BjB,SA6BiB,IAAD,IAAa,aA7BvD,IAAmC,O;MACnC,6B;IACJ,C;;M9I9HA,IAAI,E8I
oIQ,KAAK,C9Iplb,CAAJ,C;QACI,c8ImIkB,wC;Q9IIIB,MAAM,gCAAyB,OAAQ,WAAjC,C;;MAFV,IAAI,E8IqI
Q,KAAK,S9Irlb,CAAJ,C;QACI,gB8IoIqB,wEAA8D,S;Q9InInF,MAAM,gCAAyB,SAAQ,WAAjC,C;;M8IqIN,IA
AI,IAAI,CAAR,C;QACI,YAAy,iB;QACZ,UAgBsC,CAhB5B,KAgB6B,GAhBf,CAGBe,IAAD,IAAa,a;QAdnD,IA
AI,QAAQ,GAAZ,C;UACW,OAAP,aAAO,EAak,IAAL,EAaw,KAAX,EAakB,aAaIB,C;UACA,OAAP,aAAO,E
AAK,IAAL,EAaw,CAAX,EAac,GAAd,C;;UAEA,OAAP,aAAO,EAak,IAAL,EAaw,KAAX,EAakB,GAAIB,
C;;QAGX,oBAAa,G;QACb,wBAAQ,CAAR,I;IAER,C;;MAI8C,QAAC,YAAO,CAAP,IAAD,IAAa,a;IAAb,C;;M
CjL9C,QAAQ,I;MACR,QAAQ,K;MACR,YAAy,kBAAM,CAAC,OAAO,KAAP,IAAD,IAAiB,CAAjB,IAAN,C;
MACZ,OAAO,KAAK,CAAZ,C;QACI,O1J8B4E,0B0J9BrE,kBAAM,CAAN,C1J8K2B,KAAL,GAAiB,GAhJ8B,E
0J9B1D,K1J8KgB,KAAL,GAAiB,GAhJ8B,C0J9BrE,IAAP,C;UACI,a;,,;QACJ,O1J4B4E,0B0J5BrE,kBAAM,CAA
N,C1J4K2B,KAAL,GAAiB,GAhJ8B,E0J5B1D,K1J4KgB,KAAL,GAAiB,GAhJ8B,C0J5BrE,IAAP,C;UACI,a;,,;QA

CJ,IAAI,KAAK,CAAT,C;UACI,UAAU,kBAAM,CAAN,C;UACV,kBAAM,CAAN,EAAW,kBAAM,CAAN,CAA X,C;UACA,kBAAM,CAAN,EAAW,GAAX,C;UACA,a;UACA,a;;MAGR,OAAO,C;IACX,C;;MAKI,YAAY,aAA U,KAAV,EAAiB,IAAjB,EAAuB,KAAvB,C;MACZ,IAAI,QAAO,QAAQ,CAAR,IAAP,CAAJ,C;QACI,UAAU,KA AV,EAAiB,IAAjB,EAAuB,QAAQ,CAAR,IAAvB,C;MACJ,IAAI,QAAQ,KAAZ,C;QACI,UAAU,KAAV,EAAiB, KAAjB,EAAwB,KAAxB,C;IACR,C;;MAMI,QAAQ,I;MACR,QAAQ,K;MACR,YAAY,kBAAM,CAAC,OAAO,K AAP,IAAD,IAAiB,CAAjB,IAAN,C;MACZ,OAAO,KAAK,CAAZ,C;QACI,OxJK6E,0BwJLte,kBAAM,CAAN,Cx J8I2B,KAAL,GAAiB,KAZI+B,EwJL3D,KxJ8IgB,KAAL,GAAiB,KAZI+B,CwJLte,IAAP,C;UACI,a;;QACJ,OxJG 6E,0BwJHte,kBAAM,CAAN,CxJ4I2B,KAAL,GAAiB,KAZI+B,EwJH3D,KxJ4IgB,KAAL,GAAiB,KAZI+B,CwJH te,IAAP,C;UACI,a;;QACJ,IAAI,KAAK,CAAT,C;UACI,UAAU,kBAAM,CAAN,C;UACV,kBAAM,CAAN,EAA W,kBAAM,CAAN,CAAX,C;UACA,kBAAM,CAAN,EAAW,GAAX,C;UACA,a;UACA,a;;MAGR,OAAO,C;IAC X,C;;MAKI,YAAY,aAAU,KAAV,EAAiB,IAAjB,EAAuB,KAAvB,C;MACZ,IAAI,QAAO,QAAQ,CAAR,IAAP,C AAJ,C;QACI,YAAU,KAAV,EAAiB,IAAjB,EAAuB,QAAQ,CAAR,IAAvB,C;MACJ,IAAI,QAAQ,KAAZ,C;QAC I,YAAU,KAAV,EAAiB,KAAjB,EAAwB,KAAxB,C;IACR,C;;MAMI,QAAQ,I;MACR,QAAQ,K;MACR,YAAY,k BAAM,CAAC,OAAO,KAAP,IAAD,IAAiB,CAAjB,IAAN,C;MACZ,OAAO,KAAK,CAAZ,C;QACI,OzJpB8D, Yy JoBvD,kBAAM,CAAN,CzJpBwE,KAAjB,EyJoB5C,KzJpByE,KAA7B,CyJoBvD,IAAP,C;UACI,a;;QACJ,OzJtB8 D,YyJsBvD,kBAAM,CAAN,CzJtBwE,KAAjB,EyJsB5C,KzJtByE,KAA7B,CyJsBvD,IAAP,C;UACI,a;;QACJ,IAA I,KAAK,CAAT,C;UACI,UAAU,kBAAM,CAAN,C;UACV,kBAAM,CAAN,EAAW,kBAAM,CAAN,CAAX,C;UA CA,kBAAM,CAAN,EAAW,GAAX,C;UACA,a;UACA,a;;MAGR,OAAO,C;IACX,C;;MAKI,YAAY,aAAU,KAA V,EAAiB,IAAjB,EAAuB,KAAvB,C;MACZ,IAAI,QAAO,QAAQ,CAAR,IAAP,CAAJ,C;QACI,YAAU,KAAV,EA AiB,IAAjB,EAAuB,QAAQ,CAAR,IAAvB,C;MACJ,IAAI,QAAQ,KAAZ,C;QACI,YAAU,KAAV,EAAiB,KAAjB, EAAwB,KAAxB,C;IACR,C;;MAMI,QAAQ,I;MACR,QAAQ,K;MACR,YAAY,kBAAM,CAAC,OAAO,KAAP,IA AD,IAAiB,CAAjB,IAAN,C;MACZ,OAAO,KAAK,CAAZ,C;QACI,O1H7C+D,a0H6CxD,kBAAM,CAAN,C1H7C 0E,KAAiB,E0H6C7C,K1H7C2E,KAA9B,C0H6CxD,IAAP,C;UACI,a;;QACJ,O1H/C+D,a0H+CxD,kBAAM,CAA N,C1H/C0E,KAAiB,E0H+C7C,K1H/C2E,KAA9B,C0H+CxD,IAAP,C;UACI,a;;QACJ,IAAI,KAAK,CAAT,C;UA CI,UAAU,kBAAM,CAAN,C;UACV,kBAAM,CAAN,EAAW,kBAAM,CAAN,CAAX,C;UACA,kBAAM,CAAN,E AAW,GAAX,C;UACA,a;UACA,a;;MAGR,OAAO,C;IACX,C;;MAKI,YAAY,aAAU,KAAV,EAAiB,IAAjB,EAA uB,KAAvB,C;MACZ,IAAI,QAAO,QAAQ,CAAR,IAAP,CAAJ,C;QACI,YAAU,KAAV,EAAiB,IAAjB,EAAuB,Q AAQ,CAAR,IAAvB,C;MACJ,IAAI,QAAQ,KAAZ,C;QACI,YAAU,KAAV,EAAiB,KAAjB,EAAwB,KAAxB,C;I AC R,C;;MAQ+C,UAAU,KAAV,EAAiB,CAAjB,EAAoB,KAAM,KAAN,GAAa,CAAb,IAApB,C;IAAA,C;;MAE A,YAAU,KAAV,EAAiB,CAAjB,EAAoB,KAAM,KAAN,GAAa,CAAb,IAApB,C;IAAA,C;;MAEA,YAAU,KAAV ,EAAiB,CAAjB,EAAoB,KAAM,KAAN,GAAa,CAAb,IAApB,C;IAAA,C;;MAEA,YAAU,KAAV,EAAiB,CAAjB, EAAoB,KAAM,KAAN,GAAa,CAAb,IAApB,C;IAAA,C;;M5I1IuD,4C;IAAtG,C;IAAA,+D;MAC4C,+BAAW,CA AX,EAAc,CAAd,C;IAAA,C;IAD5C,oF;;MHsBI,IAAI,EEjBI,SAAU,OAAV,GAAiB,CfIBrB,CAAJ,C;QACI,cAda, qB;QAeb,MAAM,gCAAYB,OAAQ,WAAjC,C;;MEIBV,OAAO,oBAAoB,CAApB,EAAuB,CAAvB,EAA0B,SAI1 B,C;IACX,C;;MAGe,Q;MAAX,wBAAW,SAAX,gB;QAAW,SAAS,SAAX,M;QACI,SAAS,GAAG,CAAH,C;QA CT,SAAS,GAAG,CAAH,C;QACT,WAAW,cAAc,EAAc,EAakB,EAAiB,C;QACX,IAAI,SAAQ,CAAZ,C;UAAe, OAAO,I;;MAE1B,OAAO,C;IACX,C;sGAEA,yB;MAAA,8D;MAAA,iC;QASI,OAAO,cAAc,SAAS,CAAT,CAAd, EAA2B,SAAS,CAAT,CAA3B,C;MACX,C;KAVA,C;;MAqBI,OAAO,UAAW,SAAQ,SAAS,CAAT,CAAR,EAAq B,SAAS,CAAT,CAArB,C;IACtB,C;;MAuBY,Q;MALR,IAAI,MAAM,CAAV,C;QAAa,OAAO,C;MACpB,IAAI,S AAJ,C;QAAe,OAAO,E;MACtB,IAAI,SAAJ,C;QAAe,OAAO,C;MAGtB,OAA8B,iBAAtB,mDAAsB,EAAU,CAA V,C;IACIC,C;;MAYsB,uB;QAAU,2BAAoB,CAApB,EAAuB,CAAvB,EAA0B,iBAA1B,C;MAAV,C;IAAA,C;;MF vDIB,IAAI,EEsDI,SAAU,OAAV,GAAiB,CfIDrB,CAAJ,C;QACI,cAda,qB;QAeb,MAAM,gCAAYB,OAAQ,WAAj C,C;;MEqDV,sCAAkB,2BAAiB,C;IACJ,C;0FAGA,yB;MAAA,sC;MCjFA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C; MAAA,6D;QAC4C,+BAAW,CAAX,EAAc,CAAd,C;MAAA,C;MAD5C,kF;MDiFA,uBAOe,yB;QArEf,8D;;UAqE e,uB;YAAU,eAAsB,gB;YAAtB,OA5Dd,cAAc,SA4DgB,CA5DhB,CAAd,EAA2B,SA4DM,CA5DN,CAA3B,C;UA 4DI,C;QAAA,C;OAAA,C;MAPf,2B;QAOI,oCAAW,0BAAX,C;MAAA,C;KAPJ,C;0FASA,yB;MC1FA,kC;MAA A,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAc,CAAd,C;MAAA,C;MAD5C,kF;;QDkGe ,uB;UAAU,iBAAsB,kB;UAAtB,eAAkC,gB;UAAIC,OA1Dd,UAAW,SAAQ,SA0DW,CA1DX,CAAR,EAAqB,SA0

DC,CA1DD,CAArB,C;QA0DP,C;MAAA,C;MARf,uC;QAQI,oCAAW,sCAAX,C;MAAA,C;KARJ,C;4GAUA,yB; MAAA,sC;MCpGA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAc,CAAd,C; MAAA,C;MAD5C,kF;MDoGA,iCAOe,yB;QAxFf,8D;;UAWFe,uB;YAAU,eAAsB,gB;YAAAtB,OA/Ed,cAAc,SA+E gB,CA/EhB,CAAd,EAA2B,SA+EM,CA/EN,CAA3B,C;UA+EI,C;QAAA,C;OAAA,C;MAPf,2B;QAOI,oCAAW,o CAAX,C;MAAA,C;KAPJ,C;8GASA,yB;MC7GA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BA AW,CAAX,EAAc,CAAd,C;MAAA,C;MAD5C,kF;;QDuHe,uB;UAAU,iBAAsB,kB;UAAAtB,eAAkC,gB;UAAIC,O A/Ed,UAAW,SAAQ,SA+EW,CA/EX,CAAR,EAAqB,SA+EC,CA/ED,CAArB,C;QA+EP,C;MAAA,C;MAVf,uC;Q AUI,oCAAW,gDAAX,C;MAAA,C;KAVJ,C;kFAYA,yB;MAAA,sC;MCzHA,kC;MAAA,oC;;QAAsG,4C;MAAtG, C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAc,CAAd,C;MAAA,C;MAD5C,kF;MDyHA,oBAQe,yB;QA9Gf,8D;;U A8Ge,uB;YACP,sBAAsB,WAAY,SAAQ,CAAR,EAAW,CAAX,C;YACIC,Q;YAAA,IAAI,oBAAmB,CAAvB,C;c AAA,OAA0B,e;;cAAqB,eAAsB,gB;cAArE,OAvgG,cAAc,SAuG8C,CAvG9C,CAAd,EAA2B,SAuGoC,CAvGpC, CAA3B,C;;YAsGH,W;UADO,C;QAAA,C;OAAA,C;MARf,sC;QAQI,oCAAW,kCAAX,C;MAAA,C;KARJ,C;oFA aA,yB;MCtIA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAc,CAAd,C;MAA A,C;MAD5C,kF;;QD8Ie,uB;UACP,sBAAsB,WAAY,SAAQ,CAAR,EAAW,CAAX,C;UACIC,Q;UAAA,IAAI,oBA AmB,CAAvB,C;YAAA,OAA0B,e;;YAAqB,iBAAsB,kB;YAAAtB,eAAkC,gB;YAAJf,OAxGG,UAAW,SAAQ,SAw GyC,CAXGzC,CAAR,EAAqB,SAwG+B,CAXG/B,CAArB,C;;UAuGd,W;QADO,C;MAAA,C;MARf,kD;QAQI,oC AAW,8CAAX,C;MAAA,C;KARJ,C;sGAaA,yB;MAAA,sC;MCnJA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAA A,6D;QAC4C,+BAAW,CAAX,EAAc,CAAd,C;MAAA,C;MAD5C,kF;MDmJA,8BAQe,yB;QAxIf,8D;;UAWIe,uB; YACP,sBAAsB,qBAAsB,SAAQ,CAAR,EAAW,CAAX,C;YAC5C,Q;YAAA,IAAI,oBAAmB,CAAvB,C;cAAA,O AA0B,e;;cAAqB,eAAsB,gB;cAArE,OAjIG,cAAc,SAiI8C,CAjI9C,CAAd,EAA2B,SAiIoC,CAjIpC,CAA3B,C;;YA gIH,W;UADO,C;QAAA,C;OAAA,C;MARf,sC;QAQI,oCAAW,4CAAX,C;MAAA,C;KARJ,C;wGAaA,yB;MChK A,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CAAX,EAAc,CAAd,C;MAAA,C;MAD5C, kF;;QDwKe,uB;UACP,sBAAsB,qBAAsB,SAAQ,CAAR,EAAW,CAAX,C;UAC5C,Q;UAAA,IAAI,oBAAmB,CA AvB,C;YAAA,OAA0B,e;;YAAqB,iBAAsB,kB;YAAAtB,eAAkC,gB;YAAJf,OAIIG,UAAW,SAAQ,SAkIyC,CAIIZ C,CAAR,EAAqB,SAkI+B,CAII/B,CAArB,C;;UAIId,W;QADO,C;MAAA,C;MARf,kD;QAQI,oCAAW,wDAAX,C ;MAAA,C;KARJ,C;kGAcA,yB;MC9KA,kC;MAAA,oC;;QAAsG,4C;MAAtG,C;MAAA,6D;QAC4C,+BAAW,CA AX,EAAc,CAAd,C;MAAA,C;MAD5C,kF;;QDqLe,uB;UACP,sBAAsB,mBAAoB,SAAQ,CAAR,EAAW,CAAX,C ;UAA1C,OACI,oBAAmB,CAAvB,GAA0B,eAA1B,GAA+C,mBAAW,CAAX,EAAc,CAAd,C;QAFxC,C;MAAA, C;MAPf,wC;QAOI,oCAAW,4CAAX,C;MAAA,C;KAPJ,C;;MAmBe,uB;QACP,sBAAsB,SAAU,SAAQ,CAAR,EA AW,CAAX,C;QAAhC,OACI,oBAAmB,CAAvB,GAA0B,eAA1B,GAA+C,kBAAW,SAAQ,CAAR,EAAW,CAAX, C;MAFnD,C;IAAA,C;;MAAX,sCAAW,kCAAX,C;IAAA,C;;MAYc,uB;QACV,sBAAsB,mBAAoB,SAAQ,CAAR, EAAW,CAAX,C;QAA1C,OACI,oBAAmB,CAAvB,GAA0B,eAA1B,GAA+C,kBAAW,SAAQ,CAAR,EAAW,CA AX,C;MAFhD,C;IAAA,C;;MAAd,sCAAc,4CAAd,C;IAAA,C;;MAaW,uB;QAEH,UAAM,CAAN,C;UADJ,OACe, C;aACX,c;UAFJ,OAEiB,E;aACb,c;UAHJ,OAGiB,C;;UAHjB,OAIY,kBAAW,SAAQ,CAAR,EAAW,CAAX,C;MA LhB,C;IAAA,C;;MAAX,sCAAW,6BAAX,C;IAAA,C;4FASJ,yB;MAAA,4D;MAAA,wD;MAAA,mB;QAOqE,kB AAW,cAAX,C;MAAA,C;KAPrE,C;;MAGBe,uB;QAEH,UAAM,CAAN,C;UADJ,OACe,C;aACX,c;UAFJ,OAEiB, C;aACb,c;UAHJ,OAGiB,E;;UAHjB,OAIY,kBAAW,SAAQ,CAAR,EAAW,CAAX,C;MALhB,C;IAAA,C;;MAAX, sCAAW,4BAAX,C;IAAA,C;0FASJ,yB;MAAA,4D;MAAA,sD;MAAA,mB;QAOoE,iBAAU,cAAV,C;MAAA,C;K APpE,C;;MAc4F,Q;MAA7B,OAA6B,4F;IAA7B,C;;MAO6B,Q;MAA7B,OAA6B,4F;IAA7B,C;;MAOA,IAEJ,IAFI ,EAGJ,M;MAFvD,kBAD2D,SAC3D,sB;QADqD,OAC5B,SAAK,W;WAC9B,WAF2D,SAE3D,wC;QAFqD,OAE E,4F;WACvD,WAH2D,SAG3D,wC;QAHqD,OAGE,gG;;QAHF,OAI7C,uBAAmB,SAAnB,C;IAJ6C,C;;MAQrB,4B; IAAD,C;;MACS,OAAA,eAAW,SAAQ,CAAR,EAAW,CAAX,C;IAAX,C;;MAER,sB;IAAA,C;;MAGpC,sC;IAA A,C;;MACwE,OAAE,iBAAF,CAAE,EAAU,CAAV,C;IAAF,C;;MAEtB,2C;IAAA,C;;MAHID,6C;QAAA,4B;;M AAA,sC;IAAA,C;;MAMA,sC;IAAA,C;;MACwE,OAAE,iBAAF,CAAE,EAAU,CAAV,C;IAAF,C;;MAEtB,2C;IA AA,C;;MAHID,6C;QAAA,4B;;MAAA,sC;IAAA,C;;I8I/SA,C;;MA0EmE,oB;QAAA,2C;aAjB/D,kG;IAAA,C;; MAoBJ,e;MAAA,iB;MAAA,uB;IAAA,C;;MAAA,wC;MAAA,C;MASI,4E;MAMA,8E;MAOA,4E;MAOA,KE;IA7 BJ,C;;MASI,2B;MAAA,2C;IAAA,C;;MAMA,2B;MAAA,4C;IAAA,C;;MAOA,2B;MAAA,2C;IAAA,C;;MAOA ,2B;MAAA,sC;IAAA,C;;MA7BJ,+K;IAAA,C;;MAAA,a;aAAA,c;UAAA,gD;aAAA,e;UAAA,iD;aAAA,c;UAAA,

gD;aAAA,S;UAAA,2C;gBAAA,oE;;IAAA,C;;;IA8DkE,C,,,,,,,,,,,,,,,,,,,,MIGRxB,oB;QACIC,WAAW,sBAAMb,Y
AAF,CAAE,CAAnB,C;QACX,cAAM,IAAN,C;QADA,OAEA,IAAK,a;MAH6B,C;IAAA,C;;;IAIH1C,C;;iFAUA,
yB;MAAA,4B;MAAA,mC;QAMI,6BD6CQ,WC7CkB,KD6CIB,CC7CR,C;MAAA,C;KANJ,C;2GAQA,yB;MAAA
,4B;MD6CQ,kD;MC7CR,uC;QAOI,6BD2CQ,WAAO,cC3CW,SD2CX,CAAP,CC3CR,C;MAAA,C;KAPI,C;+FAU
A,yB;MAAA,kC;MAAA,mD;;QASI,sC;QAAA,4C;MATJ,C;MAAA,yF;QAWoB,2B;MAAA,CAXpB,E;MAAA,4
E;QAcY,wBAAW,MAAX,C;MAAA,C;MAdZ,sF;MAAA,sC;QASI,0D;MAAA,C;KATJ,C;;MA8BI,4BAA0D,YA
AzC,wCAA6B,UAA7B,CAAYC,CAA1D,EAAyE,yBAAzE,C;IAAA,C;;MAgBA,4BAAoE,YAAAnD,0CAA6B,QA
A7B,EAAuC,UAAvC,CAAmD,CAApE,EAAmF,yBAAnF,C;IAAA,C;;MAYyC,YAAzC,wCAA6B,UAA7B,CAAY
C,CAtEzC,oBD6CQ,WCyBsD,kBDzBtD,CC7CR,C;IAuEJ,C;;MAuD,YAAAnD,0CAA6B,QAA7B,EAAuC,UAAv
C,CAAmD,CAPFnD,oBD6CQ,WCUcGE,kBDvChE,CC7CR,C;IAqFJ,C;;MAaI,OAAc,8BAAtC,c;IAAA,C;8EAX
J,yB;MAAA,oE;MAAA,6E;;QAW0C,oB;UACIC,WAAW,sBAAMb,YAAF,CAAE,CAAnB,C;UACX,cAAM,IAA
N,C;UADA,OAEA,IAAK,a;QAH6B,C;MAAA,C;MAX1C,sC;QAWI,mBAAsC,8BAAtC,6B;QAAA,yD;MAAA,C;
KAXJ,C;qGAwBI,yB;MAAA,2D;MAAA,mB;QACI,MAAM,6BAAoB,0BAApB,C;MACV,C;KAFA,C;;MmG5IA,
yC;;;MAAA,2C;IAAA,C;;;MAAA,kD;QAAA,iC;;MAAA,2C;IAAA,C;;IA6BA,C;;MAKqB,Q;MADjB,OACI,QA
AQ,yCAAZ,GAAiB,0EAAjB,GAAgC,I;IADhC,C;;MAKA,OAAI,QAAQ,yCAAZ,GAAiB,mCAAjB,GAA4C,I;IA
A5C,C;;;MCpBQ,cAAc,GAAl,kBAAS,OAAQ,IAAjB,C;MACIB,IAAI,YAAy,mCAAhB,C;QADA,OACuC,O;;Q
AEnC,kBAaKb,oBAAQ,yCAAR,C;QACIB,IAAI,mBAAJ,C;UAJJ,OAI6B,oBAAgB,OAAhB,EAAyB,OAAzB,C;;
UACrB,WAAW,OAAQ,kBAAS,yCAAT,C;UAL3B,OAMY,SAAS,mCAAb,GAAoC,oBAAgB,OAAhB,EAAyB,W
AAzB,CAApC,GACI,oBAAgB,oBAAgB,IAAhB,EAAcB,OAAtB,CAAhB,EAAgD,WAAhD,C;;;IAGhB,C;;MAZJ,
OAAI,YAAy,mCAAhB,GAAuC,IAAvC,GACI,OAAQ,cAAK,IAAL,EAAW,4BAAX,C;IADZ,C;;;MAoCyB,Q;
MADrB,OACI,OAAA,IAAK,IAAL,EAAy,GAAZ,CAAJ,GAAqB,0EAArB,GAAoC,I;IADpC,C;;MAIA,iBAAU,O
AAV,EAAmB,IAAnB,C;IAAA,C;;MAGA,OAAI,OAAA,IAAK,IAAL,EAAy,GAAZ,CAAJ,GAAqB,mCAArB,G
AAgD,I;IAAhD,C;;;MC1D0C,wB;IAAD,C;;MAAC,yB;IAAA,C;;;MAEtD,qC;MAKI,4B;IALJ,C;;MAMqC,0C;IA
AA,C;;MAEWB,W;IAAA,C;;MACmB,c;IAAA,C;;MACJ,c;IAAA,C;;MACV,W;IAAA,C;;MACxB,Q;IAAA,C;;M
ACG,8B;IAAA,C;;;Mab7C,4C;QAAA,2B;;MAAA,qC;IAAA,C;;MAcBI,kB;MACA,wB;IAF0B,C;;MAQIB,Q;M
AFJ,UAAU,I;MACV,OAAO,IAAP,C;QACI,YAAA,GAAl,UAAJ,aAAy,GAAZ,W;UAAwB,W;;QACxB,WAAW,
GAAl,O;QACf,IAAI,oCAAJ,C;UACI,MAAM,I;;UAEN,OAAO,iBAAK,GAAL,C;;IAGnB,C;;MAGI,iBAAU,WA
AK,cAAK,OAAAL,EAAc,SAAd,CAAf,EAAyC,cAAzC,C;IAAA,C;;UAKO,I;MAFP,+BAAQ,GAAR,U;QAAoB,O
AAO,W;;MAC3B,cAAc,WAAK,kBAAS,GAAT,C;MAEf,gBAAy,WAAZ,C;QAAoB,W;WACpB,gBAAy,mCAA
Z,C;QAAqC,qB;;QAC7B,2BAAgB,OAAhB,EAAyB,cAAzB,C;MAHZ,W;IAKJ,C;;MAMc,IAAI,IAAJ,Q;MAHV,
UAAU,I;MACV,WAAW,C;MACX,OAAO,IAAP,C;QACU,uBAAI,OAAJ,GAAl,OAAJ,gC;QAAA,mB;UAAgC,O
AAO,I;;QAA7C,MAAM,M;QACN,mB;;IAER,C;;MAGI,+BAAI,OAAQ,IAAZ,GAAoB,OAApB,C;IAAA,C;;MAU
wB,Q;MAPxB,UAAU,O;MACV,OAAO,IAAP,C;QACI,IAAI,CAAC,gBAAS,GAAl,UAAb,CAAL,C;UAA4B,OA
AO,K;QACnC,WAAW,GAAl,O;QACf,IAAI,oCAAJ,C;UACI,MAAM,I;;UAEN,OAAO,gBAAS,0EAAT,C;;IAG
nB,C;;MAGI,gBAAS,KAAT,KAakB,yCAA4B,KAAM,SAAN,KAAGB,aAA5C,IAAsD,KAAM,eAAy,IAAZ,CA
A9E,C;IAAA,C;;MAE2B,OAAK,SAAL,WAAK,CAAL,GAA0B,SAAR,cAAQ,CAA1B,I;IAAA,C;;MAIvB,OAAI,
GnlqIoC,YAAU,CmlrIID,GAAMB,OAAQ,WAA3B,GAA6C,GAAF,UAAQ,O;IACvD,C;;MAFA,aAAM,kBAAK,
EAAL,EAAS,+BAAT,CAAN,GAEl,G;IAFJ,C;;MAQW,6B;QAAyB,Q;QAAT,iBAAS,sBAAT,EAAS,8BAAT,UA
AoB,O;QAAQ,W;MAA5C,C;IAAA,C;;MAGO,Q;MANIB,QAAQ,a;MACR,eAAe,gBAA+B,CAA/B,O;MACf,gBA
AY,CAAZ,C;MACA,kBAAK,kBAAL,EAAW,oDAAX,C;MnJZJ,IAAI,EmJaM,YAAS,CnJbf,CAAJ,C;QACI,cAd
W,e;QAeX,MAAM,6BAAsB,OAAQ,WAA9B,C;;MmJaN,OAAO,+BAAW,qDAAX,C;IACX,C;;MAGI,kD;MADq
B,wB;IAAD,C;;MACpB,oD;MACI,4B;IADJ,C;;;MAAA,2D;QAAA,0C;;MAAA,oD;IAAA,C;;MAI0C,gBAAT,a;
MhKslUrB,Q;MADhB,kBgKrlUmD,mC;MhKslUnD,wBAAGB,SAAhB,gB;QAAGB,cAAA,SAAhB,M;QAAsB,cA
AwB,W0B3sUID,c1B2sU+D,00B3sU/D,C;;MsIqHyC,OhKulU9B,W;IgKvIU8B,C;;;MC9ErC,MAAM,wBAAoB,s
EAApB,C;IAAN,C;8GA3BJ,yB;MAAA,2D;MAAA,sC;QA2BI,MAAM,6BAAoB,sEAApB,C;MAAN,C;KA3BJ,C;
;MAwC4C,4D;IAAA,C;;MAE5C,e;MAAA,iB;MAAA,uB;IAAA,C;;MAAA,6C;MAAA,C;MAK0C,oG;MAAqB,g
F;MAAW,4E;IAL1E,C;;;MAK0C,gC;MAAA,uD;IAAA,C;;;MAAqB,gC;MAAA,6C;IAAA,C;;;MAAW,gC;MAAA
,2C;IAAA,C;;MAL1E,sJ;IAAA,C;;;MAAA,a;aAAA,qB;UAAA,4D;aAAA,W;UAAA,kD;aAAA,S;UAAA,gD;gBA

UAAN,IAAoB,OAAF,CAAE,C;QACpB,MAAM,aAAW,CAAX,IAAN,IAAgC,OAAV,CAAE,KAAK,CAAG,C;Q
AChC,MAAM,aAAW,CAAX,IAAN,IAAiC,OAAAX,CAAE,KAAK,EAAI,C;QACjC,MAAM,aAAW,CAAX,IAAN,
IAAiC,OAAAX,CAAE,KAAK,EAAI,C;QACjC,0BAAY,CAAZ,I;MAGJ,gBAAGB,UAAU,UAAV,I;MACHB,SAAS
,sBAAS,YAAY,CAAZ,IAAT,C;MACT,aAAU,CAAV,MAAkB,SAAlB,M;QACI,MAAM,aAAW,CAAX,IAAN,IA
AqC,OAAf,EAAG,MAAK,IAAI,CAAJ,IAAL,CAAY,C;MAGzC,OAAO,K;IACX,C;MAvB4C,yB;QAAA,YAAi
B,C;MAAG,uB;QAAA,UAAe,KAAM,O;ANrF,0H;IA6BA,C;MAOyD,8BAAU,KAAV,EAAiB,CAAjB,EAAoB,
KAAM,OAA1B,C;IAAA,C;MAKP,8BAAU,cAAU,IAAV,CAAV,C;IAAA,C;MAGID,8B;MAO2B,iB;MAEvB,u
BAAoC,uB;MAuBpC,+C;IAhCJ,C;MAWgD,OAAA,oBAAc,kBAAS,QAAT,C;IAAd,C;MACd,OAAA,oBAAc,U
;IAAd,C;MACU,OAAA,oBAAc,iBAAQ,KAAR,C;IAAd,C;MACW,OAAA,oBAAc,iBAAQ,IAAR,EAAc,KAAc,
C;IAAd,C;MAEnB,OAAA,oBAAc,W;IAAd,C;MACW,OAAA,oBAAc,kBAAS,KAAT,C;IAAd,C;MACY,OAA
A,oBAAc,kBAAS,IAAT,EAAe,KAAf,C;IAAd,C;MAEjB,OAAA,oBAAc,c;IAAd,C;MAEF,OAAA,oBAAc,a;IA
Ad,C;MACa,OAAA,oBAAc,oBAAW,KAAAX,C;IAAd,C;MACc,OAAA,oBAAc,oBAAW,IAAX,EAAiB,KAAjB,
C;IAAd,C;MAE7B,OAAA,oBAAc,Y;IAAd,C;MAEoB,OAAA,oBAAc,mBAAU,KAAV,C;IAAd,C;MACP,OAA
A,oBAAc,mBAAU,IAAV,C;IAAd,C;MACqC,OAAA,oBAAc,mBAAU,KAAV,EAAiB,SAAjB,EAA4B,OAA5B,
C;IAAd,C;MA9BxF,qC;QAAA,oB;MAAA,8B;IAAA,C;MAsCA,gC;MAC0B,iB;IAD1B,C;MAEgD,OAAA,4B
AAQ,kBAAS,QAAT,C;IAAR,C;MAFhD,uC;QAAA,sB;MAAA,gC;IAAA,C;MAiBmC,yBAAa,IAAb,EAAmB
,IAAK,IAAI,EAA5B,C;IAAA,C;MAaC,yBAAa,IAAK,QAAIB,EAA2B,IAAK,YAAI,EAAJ,CAAQ,QAAxC,C;IA
AA,C;MAcpC,IAAA,KAAM,UAAN,C;QAAMB,MAAM,gCAAYB,uCAAoC,KAA7D,C;WACzB,IAAA,KAAM,
KAAN,GAAa,UAAb,C;QAF8C,OAEB,0BAAQ,KAAM,MAAd,EAAqB,KAAM,KAAN,GAAa,CAAb,IAArB,C;
WAC9B,IAAA,KAAM,MAAN,GAAc,WAAAd,C;QAH8C,OAGf,0BAAQ,KAAM,MAAN,GAAc,CAAd,IAAR,EA
AyB,KAAM,KAA/B,IAAuC,CAAvc,I;QAHe,OAIcT,mB;IAJsC,C;MAiB9C,IAAA,KAAM,UAAN,C;QAAMB,
MAAM,gCAAYB,uCAAoC,KAA7D,C;WACzB,IAAA,KAAM,KAAN,+C;QAFiD,OAElB,2BAAS,KAAM,MAAF,
EAA5B,KAAM,aAN,yBAAqB,CAArB,EAAtB,C;WAC/B,IAAA,KAAM,MAAN,+C;QAHid,OAGjB,2BAAS,K
AAM,MAAN,8BAAc,CAAd,EAAT,EAA0B,KAAM,aAAhC,0BAAGD,CAAhD,E;QAHiB,OAIzC,oB;IAJyC,C;
MAYZ,Y3DiRiC,a2DsR5B,K3DiR4B,C2DsRjC,I;IAAA,C;MAIrC,OAAA,SAAK,KAAK,EAAL,GAAU,QAAf,G
AAyC,CAAX,CAAC,QAAD,IAAW,KAAI,E;IAA7C,C;M1JtTA,IAAI,E0JwTuD,QAAQ,I1JxT/D,CAAJ,C;QACI,
c0JuTuE,+B;Q1JtVe,MAAM,gCAAYB,OAAQ,WAAjC,C;I0JsTyC,C;M1JxTnD,IAAI,E0JyTyD,sBAAQ,IAAR,
K1JzTzD,CAAJ,C;QACI,c0JwTyE,+B;Q1JvTzE,MAAM,gCAAYB,OAAQ,WAAjC,C;I0JuT2C,C;M1JzTrD,IAAI
,E0J0T6D,QAAQ,I1J1TrE,CAAJ,C;QACI,c0JyT6E,+B;Q1JxT7E,MAAM,gCAAYB,OAAQ,WAAjC,C;I0JwT+C,
C;MAEJ,oCAA0B,IAA1B,qBAAiC,KAAjC,kB;IAAA,C;MChVpB,OrK2RE,SqK3RF,mBrK2RE,C;IqK3RF,C;M
Aww,6BAAS,WAAT,EAAa,KAAb,C;IAAA,C;MAY5C,qBAAqB,IAArB,EAA2B,KAA3B,C;MAEA,iBAAiB,Ir
K6JgB,KqK7JhB,GAAiB,W;MACiC,kBAAkB,KrK4Je,KqK5Jf,GAAkB,W;MAEpC,mBAAmB,0BAAQ,UAAR,E
AAoB,WAApB,IAAqC,W;MACxD,OrK8PmC,SqK9P5B,YrK8P4B,C;IqK7PvC,C;MAaI,IAAA,KAAM,UAAN,C
;QAAMB,MAAM,gCAAYB,uCAAoC,KAA7D,C;QACzB,IrKeKe,YqKFIE,KAAM,KrKE6E,KAAjB,EqKFrD,4B
AAK,UrKE6E,KAA7B,CqKFIE,K;UAFiD,OAElB,sBAAS,KAAM,MAAf,ErKoBsB,SqKpBA,KAAM,KrKoBI,KA
AK,GAAW,CqKpBb,WrKoBa,MAAX,IAAf,CqKpBtB,C;UAC/B,IrKCKE,YqKDIE,KAAM,MrKC6E,KAAjB,Eq
KDpD,4BAAK,UrKC4E,KAA7B,CqKDIE,K;YAHid,OrKsBI,SqKnBrB,sBrKgCsB,SqKhCb,KAAM,MrKgCiB,K
AAK,GAAAY,CqKhC1B,WrKgC0B,MAAZ,IAAf,CqKhCtB,EAA2B,KAAM,KAAjC,CrKmB+B,KAAK,GAAW,C
qKnBN,WrKmBM,MAAX,IAAf,C;YqKtBJ,OAIzC,mB;IAJyC,C;MAcd,OtlkPG,UsIIPH,oBtlkPG,C;IsIIPH,C;
MAWY,8BAAU,2BAAV,EAAe,KAAf,C;IAAA,C;MAY/C,sBAASB,IAAtB,EAA4B,KAA5B,C;MAEA,iBAAiB,I
tlgHkB,KslhHIB,8B;MACjB,kBAAkB,KtI+GiB,KsI/GjB,8B;MAElB,mBAAmB,2BAAS,UAAT,EAAqB,WAArB,
+B;MACnB,OtlqNsC,UsIrn/B,YtlqN+B,C;IsIpN1C,C;MAaI,IAAA,KAAM,UAAN,C;QAAMB,MAAM,gCAAYB,
uCAAoC,KAA7D,C;QACzB,ItI9CmE,asI8CnE,KAAM,KtI9C+E,KAAIB,EsI8CtD,6BAAM,UtI9C8E,KAA9B,CsI
8CnE,K;UAFoD,OAEPB,uBAAU,KAAM,MAAhB,EtIjCuB,UsliCA,KAAM,KtIjCK,KAAK,KAAW,C/B+J7C,UA
AW,oBAAL,CqK9HyB,WrK8HzB,MAAK,CAAL,iBAAN,C+B/J6C,MAAX,CAAhB,CsliCvB,C;UACHC,ItI/Cm
E,asI+CnE,KAAM,MtI/C+E,KAAIB,EsI+CrD,6BAAM,UtI/C6E,KAA9B,CsI+CnE,K;YAHoD,Otl/BG,UsIkCtB,u
BtIrBuB,UsIqBb,KAAM,MtIrBkB,KAAK,UAAy,C/BkJ/C,UAAW,oBAAL,CqK7Hc,WrK6Hd,MAAK,CAAL,iB
AAN,C+BIJ+C,MAAZ,CAAhB,CsIqBvB,EAA4B,KAAM,KAAIC,CtIICiC,KAAK,KAAW,C/B+J7C,UAAW,oBA

AL,CqK7HgC,WrK6HhC,MAAK,CAAL,iBAAN,C+B/J6C,MAAX,CAAhB,C;;;YsI+BH,OAI5C,oB;;;IAJ4C,C;;M
AepD,4BAAU,KpIgkFH,QoIhkFP,C;MACA,OAAO,K;IACX,C;;MAOsD,OpI+nF3C,eoI/nF2C,4BAAU,IAAV,CpI
+nF3C,C;IoI/nF2C,C;;MASN,yB;QAAA,YAAiB,C;MAAG,uB;QAAA,UAAe,KAAM,K;MACrF,4BAAU,KpI6iF
H,QoI7iFP,EAA+B,SAA/B,EAA0C,OAA1C,C;MACA,OAAO,K;IACX,C;;M3JIHL,IAAI,EV0B8D,YqK4FD,KrK5
FkB,KAAjB,EqK4FO,IrK5FsB,KAA7B,CqK4FD,I3JtH7D,CAAJ,C;QACI,c2JqH6E,+B;Q3JpH7E,MAAM,gCAAy
B,OAAQ,WAAjC,C;;I2JoH+C,C;;M3JtHzD,IAAI,EqBkC+D,asIsFC,KIttFiB,KAAiB,EsIsFS,IttFqB,KAA9B,CsIs
FC,I3JxHhE,CAAJ,C;QACI,c2JuHgF,+B;Q3JtHhF,MAAM,gCAAyB,OAAQ,WAAjC,C;;I2JsHkD,C;;MCRt5D,iB;
MANA,Y;MACA,Y;MACA,Y;MACA,Y;MACA,Y;MACA,sB;M5JcA,IAAI,E4JPQ,CAAC,WAAK,QAAL,GAAU
,QAAV,GAAe,QAaf,GAAoB,QAARb,MAA2B,C5JOnC,CAAJ,C;QACI,c4JRwC,wD;Q5JSxC,MAAM,gCAAyB,
OAAQ,WAAjC,C;;MIgHV,iBAAc,CAAd,UwJtHW,ExJsHX,U;QwJtHiB,c;;IAhBrB,C;;MAoBQ,QAAQ,Q;MACR,
IAAI,IAAO,MAAO,C;MACIB,WAAI,Q;MACJ,WAAI,Q;MACJ,WAAI,Q;MACJ,SAAS,Q;MACT,WAAI,E;MAC
J,IAAK,IAAO,KAAM,CAAd,GAAsB,EAAtB,GAA8B,MAAO,C;MACzC,WAAI,C;MACJ,gCAAU,MAAV,I;MA
CA,OAAO,IAAI,aAAJ,I;IACX,C;;MAGI,OAAU,cAAV,cAAU,EAAc,QAAd,C;IAAV,C;;;MAZBJ,sD;MACQ,yBA
AK,KAAL,EAAY,KAAZ,EAAMB,CAAnB,EAAsB,CAAtB,EAA+B,CAAN,KAAzB,EAAuC,SAAU,EAAX,GAA
oB,UAAW,CAArE,C;MADR,Y;IAAA,C;;;MCL2C,qCAAiB,UAAjB,EAAwB,KAAxB,KAAkC,8BAAiB,KAAjB,
EAAwB,iBAAXB,C;IAAIC,C;;MACT,QAAC,8BAAiB,UAAjB,EAAwB,iBAAXB,C;IAAD,C;;MAYIC,4B;MACA
,0C;IAFiD,C;;MACjD,2B;IAAA,C;;MACA,kC;IAAA,C;;MAII,OAAO,0CAAgC,kBAAa,KAAM,UAAAnB,KAC/B,
mBAAS,KAAM,MAAf,KAAwB,0BAAgB,KAAM,aAAtB,CADO,CAAhC,C;IAEX,C;;MAGI,OAAW,cAAJ,GAA
e,EAaf,GAAuB,MAAW,SAAN,UAAAM,CAAX,QAAqC,SAAb,iBAaA,CAArC,I;IACIC,C;;MAEKc,OAAE,UAAf
,qBAAU,iB;IAAV,C;;MAYIC,gBAAqB,K;MACrB,uBAA4B,Y;IALD,C;;MAMQ,oB;IAAA,C;;MACO,2B;IAAA,
C;;MAEqB,YAAK,C;IAAL,C;;MAEf,gBAAS,aAAT,IAAmB,SAAS,oB;IAA5B,C;;MACd,SAAE,iBAAU,oBAAZ,
C;IAAA,C;;MAG9B,OAAO,4CAA+B,kBAAa,KAAM,UAAAnB,KAC9B,kBAAU,KAAM,SAAhB,IAA0B,yBAAiB
,KAAM,gBADnB,CAAB,C;IAEX,C;;MAGI,OAAW,cAAJ,GAAe,EAaf,GAAuB,MAAY,SAAP,aAAO,CAAZ,Q
AAuC,SAAd,oBAAc,CAAvC,I;IACIC,C;;MAEKc,OAAE,aAAf,qBAAW,oB;IAAX,C;;MAUuC,2BAAgB,SAAh
B,EAAsB,IAAtB,C;IAAA,C;;MASQ,6BAAkB,SAAIb,EAAwB,IAAxB,C;IAAA,C;;MAWjF,0BAAmB,2BAAS,O
AAT,C;IAAnB,C;;MAIA,IAAI,CAAC,UAAAL,C;QAAiB,MAAM,gCAAyB,iCAA8B,IAA9B,iBAAzB,C;IAC3B,C;
6ECxGA,yB;MAAA,4F;MAAA,2B;QAOI,MAAM,mCAA8B,0EAA9B,C;MAAN,C;KAPJ,C;;MCS8D,OAAK,oB
AAL,SAAK,CAAL,GAakB,K;IAAIb,C;;MAc1B,0B;QAAA,aAAsB,K;MAcTd,IAAI,cAAQ,KAAZ,C;QAAMb,O
AAO,I;MAC1B,IAAI,CAAC,UAAAL,C;QAAiB,OAAO,K;MAExB,IvDpBiI,OAArI,MAAoD,cuDoB5C,SvDpBsE,I
AA1B,CAAmC,cAAc,YAAW,CAAX,CAAgC,CuDoB7H,KvDpB6H,OAArI,MAAoD,cuDoBtB,KvDpBgD,IAA1B
,CAAmC,cAAc,YAAW,CAAX,CAAgC,CuDoBjI,C;QAA+C,OAAO,I;MAcTd,IvDxBiI,OAArI,MAAoD,cuDwB5
C,SvDxBsE,IAA1B,CAAmC,cAAc,YAAW,CAAX,CAAgC,CuDwB7H,KvDxB6H,OAArI,MAAoD,cuDwBtB,Kv
DxBgD,IAA1B,CAAmC,cAAc,YAAW,CAAX,CAAgC,CuDwBjI,C;QAA+C,OAAO,I;MAcTd,OAAO,K;IACX,C;
;MAKyC,QAAQ,cAAA,sCAAK,cAAL,EAAoB,sCAAK,cAAzB,CAAR,6B;IAAA,C;;MCIBZ,4B;QAAA,eAAuB,
G;MACHd,wCAAsB,EAAtB,EAA0B,YAA1B,C;IAAA,C;;MAOoC,yB;QAAA,YAAoB,E;MAAI,4B;QAAA,eAA
uB,G;MhKGNf,IAAI,CgB0N+C,CAAC,QgJ5N5C,YhJ4N4C,ChB1NpD,C;QACI,cgKHiC,wC;QhKlJc,MAAM,gC
AAyB,OAAQ,WAAjC,C;;MgKHV,cAAy,gB;MAEC,yBAAS,mBAAS,YAAA,SAAU,OAAV,EAAMb,OAAM,K
AAzB,CAAT,I;MAAT,wBAAiD,kBAakB,SAAIb,C;MA0E9D,gBAAgB,iBA1ET,OA0ES,C;MIJssCT,kBAAoB,g
B;MAwQd,gB;MADb,YAAy,C;MACC,00JxhDN,01JwhDM,W;kBAAb,OAAa,cAAb,C;QAAA,sB;QA9PsB,U;Q
AAA,cA8PT,oBAAmB,cAAnB,EAAMb,sBAAnB,U;Q0Jz8CIB,kB;;YAHA,CAAC,YAAS,CAAT,IAAc,qBAAf,K
AA4C,Q1J48CG,I0J58CH,C;UAC5C,a;;UAEA,4B;UA9E+B,uB;;YjJ4EzB,kC;YAAA,wBT28CyC,IS38CzC,C;YA
AA,qB;YAAA,oB;YAAA,oB;YAAAd,gE;cACI,IiJ7EkD,CAAI,aAAH,UjJ6ErC,YT08CqC,IS18CrC,YAAK,OAAL,
EiJ7EqC,CAAG,CjJ6EtD,C;gBACI,sBAAO,O;gBAAP,wB;;YAGR,sBAAO,E;;;UiJjFH,iD;UAGI,gCAA2B,EAA3
B,C;YAHJ,2BAGqC,I;;gBACjC,IAAK,a1JmhD0C,I0JnhD1C,gBAAyB,uBAAzB,CAAL,C;YAJJ,2B1JuhDmD,IW
z3CsB,W+I1JI,0BAAuC,mBAAvC,I/I0JJ,C;;;Y+I9JzE,2BAKY,I;;UAyER,iE5JJD,yB4JIC,4B1Jy8C+C,I;;QA9PpB,
8B;UAA6C,6B;;;M0J1xChF,OAiFK,S1J0sCE,W0J1sCF,EAAO,mBAAc,kBAAd,CAAP,EAA0C,IAA1C,CACA,W
;IAzET,C;;MAiByC,gCAAc,EAAd,C;IAAA,C;;MAKT,yB;QAAA,YAAoB,E;MAM3C,Q;MALL,cAAy,gB;MIJ4I
BL,kBAAS,gB;MAyEA,U;MAAA,S0JnqBM,01JmqBN,W;MAAhB,OAAGB,gBAAhB,C;QAAgB,2B;QAAM,IU5

e6B,CAAC,QV4ehB,OU5egB,CV4e9B,C;UAAwB,WAAy,WAAI,OAAJ,C;;M0JjqBrD,kB1JkqBE,W;MASjBA,oB
AAM,iBAAa,qCAAwB,EAAXB,CAAb,C;MAqEA,U;MAAA,+B;MAAb,OAAa,gBAAb,C;QAAa,wB;QACT,aAA
Y,WOr2CpB,YPq2CkC,IOr2ClC,CPq2CoB,C;;M0JhyChB,sBAAsB,CAGjB,c1J8xCE,a0J9xCF,CAHiB,mBAGR,C
;MAED,yBAAS,mBAAS,YAAA,SAAU,OAAV,EAAMb,OAAM,KAAzB,CAAT,I;MAAT,wBAAiD,kBAAkB,SA
AIB,C;MAmC9D,gBAAGb,iBAnCT,OAmCS,C;M1JssCT,oBAAoB,gB;MAwQd,kB;MADb,YAAy,C;MACC,S0Jj
/CN,O1Ji/CM,W;MAAb,OAAa,gBAAb,C;QAAa,0B;QA9PsB,U;QAAA,cA8PT,oBAAmB,cAAnB,EAAMb,sBAA
nB,U;Q0Jz8CIB,kB;Q1J2sC2B,c0J9sC3B,CAAC,YAAS,CAAT,IAAc,qBAAf,KAA4C,Q1J48CG,M0J58CH,C1J8s
CjB,G0J7sC3B,I1J6sC2B,G0J3sC3B,oBAxCmG,Q1Ji/CpD,M0Jj/CoD,kBAwCnG,Y5JJD,yB4JIC,4B1Jy8C+C,MA
9PpB,U;UAA6C,+B;;;M0JnvChF,OA0CK,S1J0sCE,a0J1sCF,EAAO,mBAAc,kBAAd,CAAP,EAA0C,IAA1C,CAC
A,W;IA1CT,C;;MASa,qB;QAEg,IAAG,QAAH,EAAG,CAAH,C;UAEQ,IAAA,EAAG,OAAH,GAAy,cAAO,OA
AnB,C;YAHZ,OAGyC,c;;YAHZC,OAIoB,E;;UAJpB,OAoy,iBAAS,E;MARpB,C;IAAA,C;;MAFmB,sB;QAAA,S
AAiB,M;MAC7C,OAYK,eAXA,OADL,uBACK,EAAl,4BAAJ,CAWA,EAAa,IAAb,C;IAZL,C;;MAcoC,uB;;QjJet
B,gC;QAAA,gC;QAAA,mB;QAAA,kB;QAAA,kB;QAAd,0D;UACI,liJhB+C,CAAI,aAAH,UjJgBIC,iCAAK,KAA
L,EiJhBkC,CAAG,CjJgBnD,C;YACI,sBAAO,K;YAAP,wB;;;QAGR,sBAAO,E;;;MXPA,4B;M4Jb6B,OAA8C,OA
AM,EAAV,GAAC,gBAAd,GAA0B,E;IAApE,C;;MAGE,W;IAAK,C;;MACnC,uB;QAAB,wBAAS,I;MAA3B,C;I
AAA,C;;MADR,IAAA,MhJoIgd,YAAU,CgJpI1D,C;QAD4C,OACxB,wB;;QADwB,OAEPc,kC;IAFoC,C;mBAK
hD,yB;M1JosCA,+D;MAwQA,wE;M0J58CA,sF;QAKI,gBAAGb,2B;Q1JssCT,kBAAoB,gB;QAwQd,gB;QADb,Y
AAy,C;QACC,2B;QAAb,OAAa,cAAb,C;UAAa,sB;UA9PsB,U;UAAA,cA8PT,oBAAmB,cAAnB,EAAMb,sBAA
nB,U;U0Jz8CIB,kB;U1J2sC2B,c0J9sC3B,CAAC,YAAS,CAAT,IAAc,qBAAf,KAA4C,Q1J48CG,I0J58CH,C1J8sCj
B,G0J7sC3B,I1J6sC2B,G0J3sC3B,sC1Jy8C+C,I0Jz8C/C,a5JJD,yB4JIC,4B1Jy8C+C,IA9PpB,U;YAA6C,6B;;;Q0J/
sChF,OAMK,S1J0sCE,W0J1sCF,EAAO,mBAAc,kBAAd,CAAP,EAA0C,IAA1C,CACA,W;MACT,C;KAdA,C;;;;
;;;MC3BkD,oCAAA,IAAb,C;IAAA,C;;MAiBC,kB;IAArB,C;;MAE6B,OAAA,UAAM,YAAN,aAAkB,CAAIB,C;I
AAA,C;;MAEA,OAAA,UAAM,YAAN,aAAkB,CAAIB,C;IAAA,C;;MAEA,OAAA,UAAM,YAAN,aAAkB,CAA
IB,C;IAAA,C;;MAEA,OAAA,UAAM,YAAN,aAAkB,CAAIB,C;IAAA,C;;MAEA,OAAA,UAAM,YAAN,aAAkB,
CAAIB,C;IAAA,C;;MAEA,OAAA,UAAM,YAAN,aAAkB,CAAIB,C;IAAA,C;;MAEA,OAAA,UAAM,YAAN,aA
AkB,CAAIB,C;IAAA,C;;MAEA,OAAA,UAAM,YAAN,aAAkB,CAAIB,C;IAAA,C;;MAEA,OAAA,UAAM,YAA
N,aAAkB,CAAIB,C;IAAA,C;;MAEA,OAAA,UAAM,YAAN,aAAkB,EAIB,C;IAAA,C;;MAQf,OAAA,UAAM,
YAAy,iBAAQ,CAAR,EAaw,UAAM,YAAy,KAA7B,C;IAAIB,C;;6EhFjH5C,yB;MAAA,iD;MAAA,4B;QAI4C,
kBAAM,SAAN,C;MAAA,C;KAJ5C,C;+EAMA,yB;MAAA,gD;MAAA,oC;QAI+D,kBAAM,SAAN,EAAY,MAA
Z,C;MAAA,C;KAJ/D,C;+EAMA,yB;MAAA,oC;MAAA,qC;QAIqE,sBAAM,SAAN,EAAY,OAaz,C;MAAA,C;K
AJrE,C;qFCZA,yB;MAAA,yD;MAAA,gC;QAMoB,gBAAhB,oB;QAAsB,a9EkEtB,W;Q8EIEA,O9EmEO,S8EnE8
B,W;MAArC,C;KANJ,C;uFAQA,yB;MAAA,iE;MAAA,0C;QAO4B,gBAAXB,mBAAC,QAAd,C;QAA8B,a9EyD9
B,W;Q8EzDA,O9E0DO,S8E1DsC,W;MAA7C,C;KAPJ,C;;MAaiB,Q;MAAb,wBAAa,KAAb,gB;QAAa,WAAA,K
AAb,M;QACI,yBAAO,IAAP,C;;MACJ,OAAO,S;IACX,C;;MAMiB,Q;MAAb,wBAAa,KAAb,gB;QAAa,WAAA
,KAAb,M;QACI,yBAAO,IAAP,C;;MACJ,OAAO,S;IACX,C;;MAKQ,sB;QAAqB,yBAAO,UAAU,OAAV,CAAP,
C;WACrB,sD;QAA4B,yBAAO,OAAP,C;WAC5B,2B;QAAMb,yBAAO,kBAAP,C;;QACX,yBAAe,SAAR,OAQ,
CAAf,C;IAEHb,C;;Mtf9C0C,iCAAqB,EAARb,C;IAAA,C;;MAUvB,Q;MAAA,qBAAL,SAAK,EAAY,KAAZ,C;M
AAL,iB;QAA2B,OAAO,I;;MAA5C,UAAU,I;MACV,IAAI,MAAM,sCAAK,UAAx,IAAwB,MAAM,sCAAK,UA
AvC,C;QAAkD,OAAO,I;MACzD,OAAW,OAAJ,GAAL,C;IACf,C;;MAO4C,kCAAsB,EAATb,C;IAAA,C;;MAUz
B,Q;MAAA,qBAAL,SAAK,EAAY,KAAZ,C;MAAL,iB;QAA2B,OAAO,I;;MAA5C,UAAU,I;MACV,IAAI,MAA
M,uCAAM,UAAZ,IAAyB,MAAM,uCAAM,UAAzC,C;QAAoD,OAAO,I;MAC3D,OAAW,QAAJ,GAAL,C;IACf,C
;;MAOwC,gCAAOB,EAAPb,C;IAAA,C;;MA0CIB,Q;MAhCIB,WAAW,KAAx,C;MAEA,aAAa,SAAK,O;MACIB,
IAAI,WAAU,CAAd,C;QAAiB,OAAO,I;MAExB,S;MACA,c;MACA,S;MAEA,gBAAGb,qBAAK,CAAL,C;MACH
B,IAAI,YAAy,EAAb,C;QACI,IAAI,WAAU,CAAd,C;UAAiB,OAAO,I;QAExB,QAAQ,C;QAER,IAAI,cAAa,E
AAjB,C;UACI,aAAa,I;UACb,QAAQ,W;;cACL,IAAI,cAAa,EAajB,C;UACH,aAAa,K;UACb,QAAQ,W;;;UAER,
OAAO,I;;QAEX,QAAQ,C;QACR,aAAa,K;QACb,QAAQ,W;;MAIZ,qBAAQB,QAAQ,KAAR,I;MACrB,aAAa,C;
MACK,gBAAS,CAAT,I;MAAIB,aAAU,KAAV,iB;QACI,YAAy,QAAQ,qBAAK,CAAL,CAAR,EAaiB,KAAjB,

C;QAEZ,IAAI,QAAQ,CAAZ,C;UAAe,OAAO,I;QACtB,IAAI,SAAS,cAAb,C;UAA6B,OAAO,I;QAEpC,6BAAU, KAAV,C;QAEA,IAAI,UAAS,QAAQ,KAAR,IAAT,CAAJ,C;UAA4B,OAAO,I;QAEEnC,kBAAU,KAAV,I;;MAGJ, OAAW,UAAJ,GAAgB,MAAhB,GAA4B,CAAC,MAAD,I;IACvC,C;;MAO0C,iCAAqB,EAARb,C;IAAA,C;;MA0 CpB,Q;MAhCIB,WAAW,KAAx,C;MAEA,aAAa,SAAK,O;MACIB,IAAI,WAAU,CAAd,C;QAAiB,OAAO,I;MA ExB,S;MACA,c;MACA,S;MAEA,gBAAgB,qBAAK,CAAL,C;MACHb,IAAI,YAAY,EAAbB,C;QACI,IAAI,WAA U,CAAd,C;UAAiB,OAAO,I;QAExB,QAAQ,C;QAER,IAAI,cAAa,EAAbB,C;UACI,aAAa,I;UACb,gC;;cACG,IAA I,cAAa,EAAbB,C;UACH,aAAa,K;UACb,6B;;;UAEA,OAAO,I;;QAEZ,QAAQ,C;QACR,aAAa,K;QACb,6B;;MAI J,qBAAqB,8BAAQ,KAAR,E;MACrB,e;MACkB,gBAAS,CAAT,I;MAAIB,aAAU,KAAV,iB;QACI,YAAY,QAAQ ,qBAAK,CAAL,CAAR,EAAbB,KAAjB,C;QAEZ,IAAI,QAAQ,CAAZ,C;UAAe,OAAO,I;QACtB,IAAI,uBAAS,cA AT,KAAJ,C;UAA6B,OAAO,I;QAEpC,6CAAU,KAAV,E;QAEA,IAAI,uBAAS,8BAAQ,KAAR,EAAT,KAAJ,C;U AA4B,OAAO,I;QAEEnC,6CAAU,KAAV,E;;MAGJ,OAAW,UAAJ,GAAgB,MAAhB,GAA6B,MAAD,a;IACvC,C;; MAGyD,MAAM,0BAASb,6BAA0B,KAA1B,MAAtB,C;IAAN,C;uEoBtKzD,yB;MAAA,oC;MAAA,uC;QAIi,IBa AiB,C;QACjB,eAAe,mBAAS,CAAT,I;QACf,iBAAbB,K;QAEjB,OAAO,cAAc,QAARb,C;UACI,YAAgB,CAAC,U AAL,GAAiB,UAAjB,GAAiC,Q;UAC7C,YAAY,UAAU,iCAAk,KAAL,EAAB,C;UAEZ,IAAI,CAAC,UAAL,C;Y ACI,IAAI,CAAC,KAAL,C;cACI,aAAa,I;;cAEb,0BAAc,CAAd,I;;YAEJ,IAAI,CAAC,KAAL,C;cACI,K;;cAEA,sB AAY,CAAZ,I;;QAIz,OAAO,8BAAY,UAAZ,EAAbB,WAAW,CAAX,IAAxB,C;MACX,C;KA1BA,C;yEA4BA,y B;MAAA,8B;MA5BA,oC;MA4BA,uC;QAIK,Q;QAAsB,kBAAtB,2D;QA5BD,iBAAbB,C;QACjB,eAAe,qBAAS,C AAT,I;QACf,iBAAbB,K;QAEjB,OAAO,cAAc,QAARb,C;UACI,YAAgB,CAAC,UAAL,GAAiB,UAAjB,GAAiC,Q ;UAC7C,YAsBwB,SAtBz,CAAU,mCAAk,KAAL,EAAB,C;UAEZ,IAAI,CAAC,UAAL,C;YACI,IAAI,CAAC,KA AL,C;cACI,aAAa,I;;cAEb,0BAAc,CAAd,I;;YAEJ,IAAI,CAAC,KAAL,C;cACI,K;;cAEA,sBAAY,CAAZ,I;;QAW Z,OAPO,gCAAY,UAAZ,EAAbB,WAAW,CAAX,IAAxB,CAOGC,W;MAAvC,C;KAJJ,C;iFAMA,yB;MAAA,mD ;MAAA,oC;MAAA,uC;QAIuB,UAAL,MAAK,EAAL,MAAK,EAAL,M;QAAK,mBAAL,SAAK,C;QAAL,mB;QA AA,kB;QAAA,kB;QAAd,0D;UACI,IAAI,CAAC,UAAU,iCAAk,KAAL,EAAB,CAAL,C;YACI,OAAO,8BAAY ,KAAZ,EAAbB,gBAAnB,C;QAEf,OAAO,E;MACX,C;KATA,C;mFAWA,yB;MAAA,8B;MAXA,mD;MAAA,oC; MAWA,uC;QAIK,Q;QAAsB,kBAAtB,2D;QAAsB,oB;;UAXJ,kC;UAAA,qBAAL,WAAK,C;UAAL,qB;UAAA,oB ;UAAA,oB;UAAAd,0D;YACI,IAAI,CAUyB,SAVxB,CAAU,mCAAk,KAAL,EAAB,CAAL,C;cACI,mBAAO,gCA AY,KAAZ,EAAbB,kBAAnB,C;cAAP,qB;;UAER,mBAAO,E;;QAOP,OAA4C,2B;MAA5C,C;KAJJ,C;6EAMA,y B;MAAA,mD;MAAA,+C;MAAA,oC;MAAA,uC;QAIkB,Q;QAAA,OAAa,SAAR,YAAL,SAAK,CAAQ,CAAb,W; QAAd,OAAC,cAAd,C;UAAc,uB;UACV,IAAI,CAAC,UAAU,iCAAk,KAAL,EAAB,CAAL,C;YACI,OAAO,8BA AY,CAAZ,EAAbB,QAAQ,CAAR,IAAf,C;;QAEf,OAAO,E;MACX,C;KATA,C;+EAWA,yB;MAAA,8B;MAXA,m D;MAAA,+C;MAAA,oC;MAWA,uC;QAIK,Q;QAAsB,kBAAtB,2D;QAAsB,kB;;UAXT,U;UAAA,SAAa,SAAR,Y AAL,WAAK,CAAQ,CAAb,W;UAAAd,OAAC,gBAAd,C;YAAc,yB;YACV,IAAI,CAUyB,SAVtB,CAAU,mCAAk, KAAL,EAAB,CAAL,C;cACI,iBAAO,gCAAY,CAAZ,EAAbB,QAAQ,CAAR,IAAf,C;cAAP,mB;;;UAER,iBAAO,E; ;QAOP,OAA0C,yB;MAA1C,C;KAJJ,C;;MA1DI,iBAAbB,C;MACjB,eAAe,mBAAS,CAAT,I;MACf,iBAAbB,K;M AEjB,OAAO,cAAc,QAARb,C;QACI,YAAgB,CAAC,UAAL,GAAiB,UAAjB,GAAiC,Q;QAC7C,YA6DgE,4BA7D 1C,iCAAk,KAAL,EA6D0C,E;QA3DhE,IAAI,CAAC,UAAL,C;UACI,IAAI,CAAC,KAAL,C;YACI,aAAa,I;;YAE b,0BAAc,CAAd,I;;UAEJ,IAAI,CAAC,KAAL,C;YACI,K;;YAEA,sBAAY,CAAZ,I;;MAkDiD,OA9CtD,8BAAY ,UAAZ,EAAbB,WAAW,CAAX,IAAxB,C;IA8CsD,C;;MAvC5D,Q;MAAsB,kBAAtB,2D;MA5BD,iBAAbB,C;MA CjB,eAAe,qBAAS,CAAT,I;MACf,iBAAbB,K;MAEjB,OAAO,cAAc,QAARb,C;QACI,YAAgB,CAAC,UAAL,GA AiB,UAAjB,GAAiC,Q;QAC7C,YAkEoD,4BA1E9B,mCAAk,KAAL,EAkE8B,E;QAhEpD,IAAI,CAAC,UAAL,C; UACI,IAAI,CAAC,KAAL,C;YACI,aAAa,I;;YAEb,0BAAc,CAAd,I;;UAEJ,IAAI,CAAC,KAAL,C;YACI,K;;YAE A,sBAAY,CAAZ,I;;MAuDqC,OAnD1C,gCAAY,UAAZ,EAAbB,WAAW,CAAX,IAAxB,CAOGC,W;IA4CU,C;; MAKiB,oB;;QA3C/C,gC;QAAA,gC;QAAL,mB;QAAA,kB;QAAA,kB;QAAd,0D;UACI,IAAI,CA0CsE,4BA1C3D ,iCAAk,KAAL,EA0C2D,EA1C1E,C;YACI,mBAAO,8BAAY,KAAZ,EAAbB,gBAAnB,C;YAAP,qB;;QAER,mB AAO,E;;MAuC2D,uB;IAAA,C;;MAhCjE,Q;MAAsB,kBAAtB,2D;MAAsB,oB;;QAXJ,kC;QAAA,wBAAL,WAA K,C;QAAL,qB;QAAA,oB;QAAA,oB;QAAd,0D;UACI,IAAI,CA+C0D,4BA/C/C,mCAAk,KAAL,EA+C+C,EA/C 9D,C;YACI,mBAAO,gCAAY,KAAZ,EAAbB,kBAAnB,C;YAAP,qB;;QAER,mBAAO,E;;MA4C+C,OArCV,2B; IAqCU,C;;MAKU,kB;;QApCID,Q;QAAA,OAAa,WAAR,yBAAQ,CAAb,W;QAAd,OAAC,cAAd,C;UAAc,uB;UA

CV,IAAI,CAMcKE,4BAnCvD,iCAAK,KAAL,EAmCuD,EAnCtE,C;YACI,iBAAO,8BAAY,CAAZ,EAAe,QAAQ,CAAR,IAAf,C;YAAP,mB;;;QAER,iBAAO,E;;;MAGCyD,qB;IAAA,C;;;MAzB/D,Q;MAAsB,kBAAtB,2D;MAAsB,kB;;QAXT,U;QAAA,SAAa,WAAR,eAAL,WAAK,CAAQ,CAAb,W;QAAd,OAAc,gBAAd,C;UAAc,yB;UACV,IAAI,CAwCsD,4BAxC3C,mCAAK,KAAL,EAWC2C,EAxC1D,C;YACI,iBAAO,gCAAY,CAAZ,EAAe,QAAQ,CAAR,IAAf,C;YAAP,mB;;;QAER,iBAAO,E;;;MAqC6C,OA9BV,yB;IA8BU,C;;;MA5FpD,iBAAiB,C;MACjB,eAAe,mBAAS,CAAT,I;MACf,iBAAiB,K;MAEjB,OAAO,cAAc,QAAR,B,C;QACI,YAAgB,CAAC,UAAAL,GAAiB,UAAjB,GAAiC,Q;QAC7C,YHrBR,uBgqB8B,iCAAK,KAAL,EHrB9B,E;QGuBQ,IAAI,CAAC,UAAAL,C;UACI,IAAI,CAAC,KAAL,C;YACI,aAAa,I;;YAEb,0BAAc,CAAd,I;;;UA EJ,IAAI,CAAC,KAAL,C;YACI,K;;YAEA,sBAAY,CAAZ,I;;;MAGF+B,OA5EpC,8BAAY,UAAZ,EAAwB,WAAW,CAAX,IAAxB,C;IA4EoC,C;yEAE/C,yB;MAAA,8B;MAAA,qC;MAAA,4B;QAI2C,Q;QAAD,OAAuB,KAAtB,2DAAsB,CAAO,W;MAA9B,C;KAJ1C,C;;;MASoD,oB;;QA1E7B,gC;QAAA,gC;QAAL,mB;QAAA,kB;QAAA,kB;QAAd,0D;UACI,IAAI,CHIDZ,uBgkDuB,iCAAK,KAAL,EHIDvB,EGkDQ,C;YACI,mBAAO,8BAAY,KA AZ,EAAMB,gBAAnB,C;YAAP,qB;;;QAER,mBAAO,E;;;MASeyC,uB;IAAA,C;mFAEpD,yB;MAAA,8B;MAAA,+C;MAAA,4B;QAIgD,Q;QAAD,OAAuB,UAAAtB,2DAAsB,CAAY,W;MAAnC,C;KAJ/C,C;;;MASKD,kB;;QApEhC,Q;QAAA,OAAa,WAAR,yBAAQ,CAAb,W;QAAd,OAAc,cAAd,C;UAAc,uB;UACV,IAAI,CHnEZ,uBgMEuB,iCAAK,KAAL,EHnEvB,EGMEQ,C;YACI,iBAAO,8BAAY,CAAZ,EAAe,QAAQ,CAAR,IAAf,C;YAAP,mB;;;QAER,iBAAO,E;;;MAGEuC,qB;IAAA,C;+EAElD,yB;MAAA,8B;MAAA,2C;MAAA,4B;QAI8C,Q;QAAD,OAAuB,QAAtB,2DAAsB,CAAU,W;MAAjC,C;KAJ7C,C;;;MAGB8C,uB;QAAA,UAAgB,E;MAO5C,Q;MANd,IAAI,SAAS,CAAb,C;QACI,MAAM,gCAAYB,oBAAiB,MAAjB,wBAAZB,C;MACV,IAAI,UAAU,SAAK,OAAAnB,C;QACI,OAAy,mBAAL,SAAK,EAAY,CAAZ,EAAe,SAAK,OAApB,C;MAEhB,SAAS,mBAAc,MAAd,C;MACK,gBAAS,SAAK,OAAAd,I;MAAd,aAAU,CAAV,iB;QACI,EAAG,gBAAO,OAAp,C;MACP,EAAG,gBAAO,SAAP,C;MACH,OAAO,E;IACX,C;;;MAWwC,uB;QAAA,UAAgB,E;MACnD,Q;MAAD,OAAuB,SAAtB,6DAAsB,EAAS,MAAT,EAaiB,OAAjB,CAA0B,W;IAAjD,C;;;MAYwC,uB;QAAA,UAAgB,E;MAQ1C,Q;MAPd,IAAI,SAAS,CAAb,C;QACI,MAAM,gCAAYB,oBAAiB,MAAjB,wBAAZB,C;MACV,IAAI,UAAU,SAAK,OAAAnB,C;QACI,OAAy,mBAAL,SAAK,EAAY,CAAZ,EAAe,SAAK,OAApB,C;MAEhB,SAAS,mBAAc,MAAd,C;MACT,EAAG,gBAAO,SAAP,C;MACW,gBAAS,SAAK,OAAAd,I;MAAd,aAAU,CAAV,iB;QACI,EAAG,gBAAO,OAAp,C;MACP,OAAO,E;IACX,C;;;MAWsC,uB;QAAA,UAAgB,E;MACjD,Q;MAAD,OAAuB,OAAAtB,6DAAsB,EAAO,MAAP,EAAe,OAAf,CAAwB,W;IAA/C,C;;;MAaA,OAAO,qBAAgB,SAAK,OAAAL,KAAe,C;IAC1C,C;;;MAQoD,4BAAU,C;IAAV,C;;;MAQG,0BAAS,C;IAAT,C;mFAMvD,yB;MAAA,2C;MAAA,4B;QAMuD,QAAC,kB;MAAD,C;KANvD,C;yFAQA,yB;MAAA,2C;MAAA,4B;QAWI,OAAO,qBAAqB,QAAL,SAAK,C;MAChC,C;KAZA,C;;;MAiB4D,kC;MAAS,uB;MACjE,eAAoB,C;IADoC,C;;;MAGb,gB;MAAA,iE;MAAJ,4C;IAAA,C;;;MAEE,sBAAQ,yB;IAAR,C;;;MALe,4C;IAAA,C;;;MAUd,uCAAQ,E;IAAR,C;;;MAW1C,OA5DgD,qBAAU,CA4D1D,GAAe,cAAf,GAAMC,S;IAAnC,C;6EA EJ,yB;MAAA,2C;MAAA,0C;QASI,OAAI,kBAAJ,GAAe,cAAf,GAAMC,S;MAAnC,C;KATJ,C;;;MAeY,uBAAG,mBAAS,CAAT,IAAH,C;IAAA,C;;;MAMA,OAAA,SAAK,OAAAL,GAAc,CAAd,I;IAAA,C;;;MAMW,Q;MAAA,0BAAS,CAAT,I;MAAnB,OAAgB,CAAT,8BACgB,gBAAZ,qBAAK,KAAL,CAAY,CADhB,IAEoB,eAAhB,qBAAK,QAAQ,CAAR,IAAL,CAAgB,C;IAC/B,C;;;MAKuD,OCpIyC,oBDol/B,KAAAM,MCpIyB,EDoIlB,KAAAM,aAAN,GAAqB,CAArB,ICpIkB,C;IDoIzC,C;;;MAKc,qCAAY,KAAAM,MAAIB,EAAYB,KAAAM,aAAN,GAAqB,CAArB,IAAZB,C;IAAA,C;;;MAWM,2BAAY,KA AZ,EAAMB,GAAAnB,C;IAAA,C;;;MASjB,wB;QAAA,WAAgB,gB;MAAkB,OAAA,8BAAY,UAAZ,EAAwB,QAAxB,CAAkC,W;IAAIC,C;;;MAK/B,OAAA,8BAAY,KAAAM,MAAIB,EAAYB,KAAAM,aAAN,GAAqB,CAArB,IAAZB,CAAiD,W;IAAjD,C;;;MAMV,qC;QAAA,wBAAgC,S;MAC/E,YAAy,sBAAQ,SAAR,C;MACZ,OAAW,UAAAS,EAAPB,GAAwB,qBAAxB,GC1K4F,oBD0K/B,CC1K+B,ED0K5B,KC1K4B,C;ID2KhG,C;;;MAMqD,qC;QAAA,wBAAgC,S;MACjF,YAAy,sBAAQ,SAAR,C;MACZ,OAAW,UAAAS,EAAPB,GAAwB,qBAAxB,GCnL4F,oBDmL/B,CCnL+B,EDmL5B,KCnL4B,C;IDoLhG,C;;;MAMkD,qC;QAAA,wBAAgC,S;MAC9E,YAAy,sBAAQ,SAAR,C;MACZ,OAAW,UAAAS,EAAPB,GAAwB,qBAAxB,GC5L4F,oBD4L/B,QAAQ,CAAR,IC5L+B,ED4LpB,gBC5LoB,C;ID6LhG,C;;;MAMoD,qC;QAAA,wBAAgC,S;MACHf,YAAy,sBAAQ,SAAR,C;MACZ,OAAW,UAAAS,EAAPB,GAAwB,qBAAxB,GCrM4F,oBDqM/B,QAAQ,SAAU,OAAIB,ICrM+B,EDqML,gBCrMK,C;ID5MhG,C;;;MAMuD,qC;QAAA,wBAAgC,S;MACnF,YAAy,0BAAY,SAAZ,C;MACZ,OAAW,UAAAS,EAAPB,GAAwB,qBAAxB,GC9M4F,oBD8M/B,CC9M+B,ED8M5B,KC9M4B,C;ID+MhG,C;;;MAMyD,qC;QAAA,wBAAgC,S;MACrF,YAAy,0BAAY,SAAZ,C;MACZ,OAAW,UAA

S,EAAPB,GAAwB,qBAAXB,GCvN4F,oBDuN/B,CCvN+B,EDuN5B,KCvN4B,C;IDwNhG,C;;MAMsD,qC;QAAA,wBAAGC,S;MACIF,YAAAY,0BAAY,SAAZ,C;MACZ,OAAW,UAAAS,EAAPB,GAAwB,qBAAXB,GChO4F,oBDgO/B,QAAQ,CAAR,IChO+B,EDgOpB,gBChOoB,C;IDiOhG,C;;MAMwD,qC;QAAA,wBAAGC,S;MACpF,YAAAY,0BAAY,SAAZ,C;MACZ,OAAW,UAAAS,EAAPB,GAAwB,qBAAXB,GCzO4F,oBDyO/B,QAAQ,SAAU,OAAIB,I CzO+B,EDyOL,gBCzOK,C;ID0OhG,C;;MASI,IAAI,WAAW,UAAf,C;QACI,MAAM,8BAA0B,gBAAa,QAAb,oCAAkD,UAAID,OAAIB,C;MACV,SAAS,sB;MACT,EAAG,gBAAO,SAAP,EAAa,CAAb,EAAGB,UAAhB,C;MAC H,EAAG,gBAAO,WAAP,C;MACH,EAAG,gBAAO,SAAP,EAAa,QAAb,EAAuB,gBAAvB,C;MACH,OAAO,E;I ACX,C;yFAEA,yB;MAAA,8B;MAAA,qD;MAAA,+D;QAOK,Q;QAAD,OAAuB,aAAtB,2DAAsB,EAAa,UAAb,E AAYB,QAAzB,EAAmC,WAAnc,CAAgD,W;MAAvE,C;KAPJ,C;;MAGBI,+BAAa,KAAM,MAAnB,EAA0B,KAA M,aAAN,GAAqB,CAArB,IAA1B,EAakD,WAAID,C;IAAA,C;yFAEJ,yB;MAAA,8B;MAAA,qD;MAAA,gD;QA OK,Q;QAAD,OAAuB,aAAtB,2DAAsB,EAAa,KAAb,EAAoB,WAAPB,CAAIc,W;MAAxD,C;KAPJ,C;;MAkBI,I AAI,WAAW,UAAf,C;QACI,MAAM,8BAA0B,gBAAa,QAAb,oCAAkD,UAAID,OAAIB,C;MAEV,IAAI,aAAY, UAAhB,C;QACI,OAAy,mBAAL,SAAK,EAAy,CAAZ,EAAe,gBAaf,C;MAEhB,SAAS,mBAAc,oBAAU,QAAV, GAAqB,UAArB,KAAc,C;MACT,EAAG,gBAAO,SAAP,EAAa,CAAb,EAAGB,UAAhB,C;MACH,EAAG,gBAAO ,SAAP,EAAa,QAAb,EAAuB,gBAAvB,C;MACH,OAAO,E;IACX,C;uFAEA,yB;MAAA,8B;MAAA,mD;MAAA,k D;QASK,Q;QAAD,OAAuB,YAAtB,2DAAsB,EAAy,UAAZ,EAAwB,QAAXB,CAAKC,W;MAAzD,C;KATJ,C;;M AgBqE,8BAAY,KAAM,MAAIB,EAAyB,KAAM,aAAN,GAAqB,CAArB,IAAZB,C;IAAA,C;uFAErE,yB;MAAA, 8B;MAAA,mD;MAAA,mC;QAOK,Q;QAAD,OAAuB,YAAtB,2DAAsB,EAAy,KAAZ,CAAmB,W;MAA1C,C;K APJ,C;;MAcI,IAAI,wBAAW,MAAX,CAAJ,C;QACI,OAAO,8BAAY,MAAO,OAAAnB,EAA2B,gBAA3B,C;;MAE X,OAAO,8BAAY,CAAZ,EAAe,gBAaf,C;IACX,C;;MAOI,IAAI,wBAAW,MAAX,CAAJ,C;QACI,OC3VyE,oBD 2VxD,MAAO,OC3ViD,C;;MD6V7E,OAAO,S;IACX,C;;MAOI,IAAI,sBAAS,MAAT,CAAJ,C;QACI,OAAO,8BA AY,CAAZ,EAAe,mBAAS,MAAO,OAAhB,IAAf,C;;MAEX,OAAO,8BAAY,CAAZ,EAAe,gBAaf,C;IACX,C;;M AOI,IAAI,sBAAS,MAAT,CAAJ,C;QACI,OC9WwF,oBD8WvE,CC9WuE,ED8WpE,mBAAS,MAAO,OAAhB,IC9 WoE,C;;MDgX5F,OAAO,S;IACX,C;;MAQI,IAAK,qBAAU,MAAO,OAAP,GAAgB,MAAO,OAAvB,IAAV,CAA D,IAA6C,wBAAW,MAAX,CAA7C,IAAmE,sBAAS,MAAT,CAAvE,C;QACI,OAAO,8BAAY,MAAO,OAAAnB,E AA2B,mBAAS,MAAO,OAAhB,IAA3B,C;;MAEX,OAAO,8BAAY,CAAZ,EAAe,gBAaf,C;IACX,C;;MAQI,IAA K,qBAAU,MAAO,OAAP,GAAgB,MAAO,OAAvB,IAAV,CAAD,IAA6C,wBAAW,MAAX,CAA7C,IAAmE,sBA AS,MAAT,CAAvE,C;QACI,OCtYwF,oBDsYvE,MAAO,OCtYgE,EDsYxD,mBAAS,MAAO,OAAhB,ICtYwD,C;; MDwY5F,OAAO,S;IACX,C;;MAOmF,oCAAKB,SAAIB,EAA6B,SAA7B,C;IAAA,C;;MAOZ,sCAAKB,SAAIB,E AA6B,SAA7B,C;IAAA,C;;MAMD,qC;QAAA,wBAAGC,S;MACIG,YAAAY,sBAAQ,SAAR,C;MACL,Q;MAAA,I AAI,UAAAS,EAAb,C;QAAA,OAAiB,qB;;QA5JvB,U;QA4JM,OA5JgB,aAAtB,+DAAsB,EA4JyC,CA5JzC,EA4J4C ,KA5J5C,EA4JmD,WA5JnD,CAAgD,W;;MA4JvE,W;IACJ,C;;MAMwE,qC;QAAA,wBAAGC,S;MACpG,YAAAY,s BAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAAS,EAAb,C;QAAA,OAAiB,qB;;QArKvB,U;QAqKM,OArKgB,aAAt B,+DAAsB,EAqKyC,CArKzC,EAqK4C,CArK5C,EAqKmD,WArKnD,CAAgD,W;;MAqKvE,W;IACJ,C;;MAMq E,qC;QAAA,wBAAGC,S;MACjG,YAAAY,sBAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAAS,EAAb,C;QAAA,OAAi B,qB;;QAA2B,iBAAa,QAAQ,CAAR,I;QAAb,eAAwB,gB;QA9K1E,U;QA8KM,OA9KgB,aAAtB,+DAAsB,EAAa, UAAb,EAAyB,QAAzB,EA8K4D,WA9K5D,CAAgD,W;;MA8KvE,W;IACJ,C;;MAMuE,qC;QAAA,wBAAGC,S; MACnG,YAAAY,sBAAQ,SAAR,C;MACL,Q;MAAA,IAAI,UAAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,Q AAQ,SAAU,OAAIB,I;QAAb,eAAuB,gB;QAvLzF,U;QAUlM,OAvLgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,Q AAzB,EAuL2E,WAvL3E,CAAgD,W;;MAuLvE,W;IACJ,C;;MAM2E,qC;QAAA,wBAAGC,S;MACvG,YAAAY,0B AAY,SAAZ,C;MACL,Q;MAAA,IAAI,UAAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,SAAU,OAAI B,I;QAAb,eAAuB,gB;QAhMzF,U;QAgMM,OAHMgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAgM2E, WAhM3E,CAAgD,W;;MAGMvE,W;IACJ,C;;MAMyE,qC;QAAA,wBAAGC,S;MACrG,YAAAY,0BAAY,SAAZ,C; MACL,Q;MAAA,IAAI,UAAAS,EAAb,C;QAAA,OAAiB,qB;;QAA2B,iBAAa,QAAQ,CAAR,I;QAAb,eAAwB,gB; QAZM1E,U;QAYMM,OAzMgB,aAAtB,+DAAsB,EAAa,UAAb,EAAyB,QAAzB,EAyM4D,WAZM5D,CAAgD,W; ;MAYMvE,W;IACJ,C;;MAM0E,qC;QAAA,wBAAGC,S;MACtG,YAAAY,0BAAY,SAAZ,C;MACL,Q;MAAA,IAAI ,UAAAS,EAAb,C;QAAA,OAAiB,qB;;QAINvB,U;QAKNM,OAInGB,aAAtB,+DAAsB,EAKNyC,CAINzC,EAKN4C, KAIN5C,EAKNmD,WAINnD,CAAgD,W;;MAKNvE,W;IACJ,C;;MAM4E,qC;QAAA,wBAAGC,S;MACxG,YAAAY

,0BAAY,SAAZ,C;MACL,Q;MAAA,IAAI,UAAS,EAAb,C;QAAA,OAAiB,qB;;QA3NvB,U;QA2NM,OA3NgB,aA
AtB,+DAAsB,EA2NyC,CA3NzC,EA2N4C,KA3N5C,EA2NmD,WA3NnD,CAAgD,W;;MA2NvE,W;IACJ,C;;MAc
oF,OAAA,KAAM,iBAAQ,SAAR,EAAC,WAAd,C;IAAN,C;+EAEPf,yB;M0F7kBI,iE;MAAA,wC;M1F6kBJ,8C;Q
AOU,yB;;U0F9kBF,Y1F8kBJ,K0F9kBgB,c1F8kBF,S0F9kBE,C;UACZ,IAAI,AAAJ,C;YAAMb,wB1F6kBT,S0F7k
BsB,W;YAAb,0B;;UAEnB,gBAAgB,C;UACHb,a1F0kBU,S0F1kBS,O;UACnB,SAAS,mBAAC,MAAd,C;;YAEI,i
BAAiB,oB;YACjB,EAAG,gB1FskBG,S0FtkBH,EAAC,SAAd,EAAYB,UAAW,MAAM,MAA1C,C;YACH,EAAG,
gB1FqkBS,S0FrkBF,CAAU,UAAV,CAAP,C;YACH,YAAAY,UAAW,MAAM,AAAJB,GAAgC,CAAhC,I;YACZ,Q
AAQ,UAAW,O;;UACd,oBAAY,MAAZ,IAAsB,AAAtB,C;UAET,IAAI,YAAAY,MAAhB,C;YACI,EAAG,gB1F+jB
G,S0F/jBH,EAAC,SAAd,EAAYB,MAAZB,C;;UAGP,wBAAO,EAAG,W;;;Q1F4jBd,4B;MAAA,C;KAPJ,C;;MAey
F,OAAA,KAAM,sBAaA,SAAb,EAAMb,WAAAnB,C;IAAN,C;;MAOnB,OAAA,KAAM,iBAAQ,SAAR,C;IAAN,C;
;MAOIE,IAAK,cAAc,CAAF,IAAsB,AAaA,CAAnC,IAA0C,cAAa,SAAK,OAAL,GAAC,MAAd,IAAb,CAA1C,IAAi
F,eAAc,KAAM,OAAN,GAaE,MAAf,IAAd,CAArF,C;QACI,OAAO,K;;MAGX,iBAAC,CAAd,UAsB,MAAtB,U;
QACI,IAAI,CAA0B,SAAZB,qBAAK,AAaA,KAAb,IAAL,CAAYB,EAAO,iBAAM,cAAc,KAAd,IAAN,CAAP,EA
AmC,UAAnc,CAA9B,C;UACI,OAAO,K;;MAEf,OAAO,I;IACX,C;;MAK+C,0B;QAAA,aAsB,K;MACjE,OAA
A,SAAK,OAAL,GAAC,CAAd,IAA2B,SAAR,qBAAK,CAAL,CAAQ,EAAO,IAAP,EAAa,UAb,C;IAA3B,C;;MA
KyC,0B;QAAA,aAsB,K;MAC/D,OAAA,SAAK,OAAL,GAAC,CAAd,IAAmC,SAAhB,qBAAK,2BAAL,CAAgB,
EAAO,IAAP,EAAa,UAb,C;IAAnC,C;;MAKqD,0B;QAAA,aAsB,K;MAC3E,IAAI,CAAC,UAAD,IAAe,6BAAf
,IAAiC,0BAArC,C;QACI,OAAAY,WAAL,SAAK,EAAW,MAAX,C;;QAEZ,OAAO,6BAakB,CAAIB,EAAqB,MA
ArB,EAA6B,CAA7B,EAAGC,MAAO,OAAvC,EAA+C,UAA/C,C;IACf,C;;MAK0E,0B;QAAA,aAsB,K;MAC5F,
IAAI,CAAC,UAAD,IAAe,6BAAf,IAAiC,0BAArC,C;QACI,OAAAY,aAAL,SAAK,EAAW,MAAX,EAAMb,UAAnc
B,C;;QAEZ,OAAO,6BAakB,UAAIB,EAA8B,MAA9B,EAAc,CAAtC,EAAYC,MAAO,OAAd,EAAwD,UAAx
D,C;IACf,C;;MAKuD,0B;QAAA,aAsB,K;MACzE,IAAI,CAAC,UAAD,IAAe,6BAAf,IAAiC,0BAArC,C;QACI,
OAAAY,SAAL,SAAK,EAAS,MAAT,C;;QAEZ,OAAO,6BAakB,mBAAS,MAAO,OAAd,IAAIB,EAA0C,MAA1
C,EAAkD,CAAlD,EAAqD,MAAO,OAA5D,EAAoE,UAApE,C;IACf,C;;MAa8D,0B;QAAA,aAsB,K;MACHf,qB
X11BO,MAAK,KW01Be,SAAK,OX11BpB,EW01B4B,KAAM,OX11BIC,C;MW41BZ,QAAQ,C;MACR,OAAO,IAA
I,cAAAJ,IAA8B,SAAR,qBAAK,CAAL,CAAQ,EAAO,iBAAM,CAAN,CAAP,EAA8B,UAA9B,CAArC,C;QACI,a;
MAEJ,IAAS,mBAAL,SAAK,EAAMb,IAAI,CAAJ,IAAnB,CAAL,IAAwC,mBAAN,KAAM,EAAMb,IAAI,CAAJ,
IAAnB,CAA5C,C;QACI,a;;MAEJ,OAAO,8BAAY,CAAZ,EAAe,CAAF,CAAKB,W;IAC7B,C;;MAU8D,0B;QAAA
,aAsB,K;MACHf,iBAAiB,SAAK,O;MACtB,kBAakB,KAAM,O;MACxB,qBXjnBO,MAAK,KWinBe,UXjnBf,E
WinB2B,WXjnB3B,C;MWmnBZ,QAAQ,C;MACR,OAAO,IAAI,cAAAJ,IAA+C,SAAZB,qBAAK,AAaA,CAAb,GA
AiB,CAAJB,IAAL,CAAYB,EAAO,iBAAM,cAAc,CAAd,GAakB,CAAIB,IAAN,CAAP,EAAGD,UAAhD,CAAtD,
C;QACI,a;;MAEJ,IAAS,mBAAL,SAAK,EAAMb,AAaA,CAAb,GAAiB,CAAJB,IAAnB,CAAL,IAAqD,mBAAN,K
AAM,EAAMb,cAAc,CAAd,GAakB,CAAIB,IAAnB,CAAzD,C;QACI,a;;MAEJ,OAAO,8BAAY,AAaA,CAAb,IA
AZ,EAA4B,UAA5B,CAAwC,W;IACnD,C;;MAaqD,0B;QAAA,aAkB,C;MAAG,0B;QAAA,aAsB,K;MAMnE,
UAAkB,M;MAL3C,IAAI,CAAC,UAAD,IAAe,KAAM,OAAN,KAAC,CAA7B,IAAkC,6BAAtC,C;QACI,WAAiB,
SAAN,KAAM,C;QACjB,OC7oBwF,kB0F7KE,oB3F0zBrE,I2F1zBqE,C1F6KF,ED6oB7D,UC7oB6D,C;;MDgpBn
E,uBAAX,UAAW,EAAC,CAAd,C;MAAkB,oC;kBAA3C,gD;QACI,kBAakB,qBAAI,KA AJ,C;QACR,c;;U7BiuSE
,U;UAAhB,4B6BjuSQ,K7BiuSR,kB;YAAgB,cAAhB,U6BjuSQ,K7BiuSR,S;YAAAsB,I6BjuSC,SAAH,U7BiuSgB,o
B6BjuShB,CAAG,0B7BiuSD,C;CAAwB,AAAO,I;cAAP,e;;UAC9C,AAAO,K;;Q6BluSH,e;UACI,OAAO,K;;MAEf
,OAAO,E;IACX,C;;MAWyD,0B;QAAA,aAkB,2B;MAAW,0B;QAAA,aAsB,K;MACxG,IAAI,CAAC,UAAD,IA
AAe,KAAM,OAAN,KAAC,CAA7B,IAAkC,6BAAtC,C;QACI,WAAiB,SAAN,KAAM,C;QACjB,OCjqB4F,sB0F7
KM,oB3F80BzE,I2F90ByE,C1F6KN,EDiqB7D,UCjqB6D,C;;kBDqqBhG,iBAAYB,eAAX,UAAW,EAAa,2BAAb,
CAAzB,WAAwD,CAAxD,U;QACI,kBAakB,qBAAI,KA AJ,C;QACR,c;;U7BysSE,Q;UAAhB,wB6BzsSQ,K7Bys
SR,gB;YAAgB,cAAhB,U6BzsSQ,K7BysSR,O;YAAAsB,I6BzsSC,SAAH,U7BysSgB,oB6BzsShB,CAAG,0B7BysS
D,C;cAAwB,AAAO,I;cAAP,e;;UAC9C,AAAO,K;;Q6B1sSH,e;UACI,OAAO,K;;MAGf,OAAO,E;IACX,C;;MAG2
G,oB;QAAA,OAAGB,K;MAOrG,UAKA,M;MAXIB,cAAkB,CAAC,IAAL,GACV,AAAW,gBAAX,UAAW,EAAC,
CAAd,CAAX,EAAc,eAAT,QAAS,EAAa,gBAAb,CAAtC,CADU,GAGV,SAAW,eAAX,UAAW,EAAa,2BAAb,
CAAX,EAAMd,gBAAT,QAAS,EAAC,CAAd,CAAnD,C;MAEJ,IAAI,iCAakB,yBAAtB,C;QACkB,yB;QAAd,OA

Ac,cAAAd,C;UAAc,uB;UACV,IAAU,cAAN,KAAM,EAAc,CAAd,EAAiB,SAAjB,EAAuB,KAAvB,EAA8B,KAA
M,OAApC,EAA4C,UAA5C,CAAV,C;YACI,OAAO,K;;;QAGD,2B;QAAd,OAAc,gBAAd,C;UAAc,2B;UACV,IA
AU,kBAAN,KAAM,EAakB,CAAIB,EAAqB,SAArB,EAA2B,OAA3B,EAakC,KAAM,OAAxC,EAAGD,UAAhD,
CAAV,C;YACI,OAAO,O;;;MAGnB,OAAO,E;IACX,C;;MAYsB,UAMA,M;MAfIB,IAAI,CAAC,UAAD,IAAe,O
AAQ,KAAR,KAAgB,CAAnC,C;QACI,aAAqB,UAR,OAAQ,C;QACrB,YAAgB,CAAC,IAAL,GAAW,sBAAQ,
MAAR,EAAGB,UAAhB,CAAX,GAA4C,0BAAy,MAAZ,EAAoB,UAApB,C;QACxD,OAAW,QAAQ,CAAZ,GA
Ae,IAAf,GAAYB,UAA5,MAAT,C;;MAGpC,cAAkB,CAAC,IAAL,GAAW,aAAW,gBAAX,UAAW,EAAc,CAAd,
CAAX,EAA6B,gBAA7B,CAAX,GAAoD,SAAW,eAAX,UAAW,EAAa,2BAAb,CAAX,EAA0C,CAA1C,C;MAEI
E,IAAI,6BAAJ,C;QACkB,yB;oBAAd,OAAc,cAAAd,C;UAAc,yB;UACmB,sB;;YVzpBrB,U;YAAA,SUypBa,OVzp
Bb,W;YAAhB,OAAgB,gBAAhB,C;cAAGB,2B;cAAM,IUypBgC,cVzpBIB,OUypBkB,EAAc,CAAd,sBVzpBIB,O
UypBmD,OAAjC,aVzpBhC,C;gBAAwB,qBAAO,O;gBAAP,uB;;;YAC9C,qBAAO,I;;;UUwpBC,uC;UACA,IAAI,s
BAAJ,C;YACI,OAAO,YAAS,cAAT,C;;;QAGD,2B;oBAAd,OAAc,gBAAd,C;UAAc,2B;UACmB,wB;;YV/pBrB,
U;YAAA,SU+pBa,OV/pBb,W;YAAhB,OAAgB,gBAAhB,C;cAAGB,6B;cAAM,IU+pBgC,kBV/pBIB,SU+pBkB,E
AAkB,CAAIB,sBV/pBIB,SU+pBuD,OAArC,aV/pBhC,C;gBAAwB,uBAAO,S;gBAAP,uB;;;YAC9C,uBAAO,I;;;U
U8pBC,2C;UACA,IAAI,wBAAJ,C;YACI,OAAO,YAAS,gBAAT,C;;MAInB,OAAO,I;IACX,C;;MAc+D,0B;QAA
A,aAAkB,C;MAAG,0B;QAAA,aAAsB,K;MACtG,4BAAU,OAAV,EAAMB,UAAAnB,EAA+B,UAA/B,EAakD,KA
AID,C;IAAA,C;;MAc+D,0B;QAAA,aAAkB,2B;MAAW,0B;QAAA,aAAsB,K;MACIH,4BAAU,OAAV,EAAMB,
UAAAnB,EAA+B,UAA/B,EAakD,IAAID,C;IAAA,C;;MAa4D,0B;QAAA,aAAkB,C;MAAG,0B;QAAA,aAAsB,K;
MACvG,gB;MAAA,8CAAU,OAAV,EAAMB,UAAAnB,EAA+B,UAA/B,EAakD,KAAID,mDAAMe,E;IAAnE,C;;
MACgE,0B;QAAA,aAAkB,2B;MAAW,0B;QAAA,aAAsB,K;MACnH,gB;MAAA,8CAAU,OAAV,EAAMB,UAAAn
B,EAA+B,UAA/B,EAakD,IAAID,mDAaKe,E;IAAIE,C;;MAWwC,0B;QAAA,aAAkB,C;MAAG,0B;QAAA,aAA
sB,K;MACnF,OAAW,cAAc,gCAAzB,GACI,sBAAW,mBAAy,IAAZ,CAAX,EAA8B,UAA9B,EAA0C,UAA1C,C
ADJ,GCryB4F,kB0F7KE,oB3Fq9B5E,I2Fr9B4E,C1F6KF,EDwyBpE,UCxyBoE,C;IDyyBhG,C;;MASgD,0B;QAA
A,aAAkB,C;MAAG,0B;QAAA,aAAsB,K;MACvF,OAAW,cAAc,gCAAzB,GACI,sBAAQ,MAAR,EAAGB,UAAh
B,EAA4B,gBAA5B,EAAoC,UAApC,CADJ,GCnzB4F,kBDszB1E,MCtzB0E,EDszBIE,UCtzBkE,C;IDuzBhG,C;;
MAUgD,0B;QAAA,aAAkB,2B;MAAW,0B;QAAA,aAAsB,K;MAC/F,OAAW,cAAc,gCAAzB,GACI,0BAAE,mB
AAy,IAAZ,CAAf,EAakC,UAAIC,EAA8C,UAA9C,CADJ,GC/zBgG,sB0F7KM,oB3F++BhF,I2F+/BgF,C1F6KN,
EDk0BpE,UCI0BoE,C;IDm0BpG,C;;MAUoD,0B;QAAA,aAAkB,2B;MAAW,0B;QAAA,aAAsB,K;MACnG,OAA
W,cAAc,gCAAzB,GACI,sBAAQ,MAAR,EAAGB,UAAhB,EAA4B,CAA5B,EAA+B,UAA/B,EAakD,IAAID,CAD
J,GC90BgG,sBDi1B1E,MCj1B0E,EDi1BIE,UCj1BkE,C;IDk1BpG,C;;MAQ+D,0B;QAAA,aAAsB,K;MACjF,OAA
I,yBAAJ,GACI,sBAAQ,KAAR,UAA4B,UAA5B,KAA2C,CAD/C,GAGI,sBAAQ,KAAR,EAAe,CAAf,EAakB,gB
AAIB,EAA0B,UAA1B,KAAyC,C;IAH7C,C;;MAakD,0B;QAAA,aAAsB,K;MACxE,6BAAQ,IAAR,UAA2B,UAA
3B,KAA0C,C;IAA1C,C;;MAMsE,OAAA,KAAM,yBAAGB,SAAhB,C;IAAN,C;;MAOtE,oB;MACA,8B;MACA,o
B;MACA,kC;IAJiC,C;;MAOa,gE;MAC1C,iBAAqB,E;MACrB,yBAAwC,WAAx,yCAAW,EAAS,CAAT,EAAY,o
CAAM,OAAIB,C;MACxC,uBAA2B,sB;MAC3B,gBAA0B,I;MAC1B,eAAmB,C;IALuB,C;;MAQtC,IAAI,uBAAk
B,CAAtB,C;QACI,iBAAy,C;QACZ,gBAAW,I;;;QAEX,IAAI,4CAAQ,CAAR,IAAa,uDAaA,yCAA1B,IAAmC,uB
AAkB,yCAAM,OAA/D,C;UACI,gBAAW,qCAAYB,iBAAN,yCAAM,CAAzB,C;UACX,uBAAkB,E;;;UAEIB,YA
AkB,iDAAN,yCAAM,EAAa,oBAAb,C;UACIB,IAAI,SAAS,IAAb,C;YACI,gBAAW,qCAAYB,iBAAN,yCAAM,C
AAzB,C;YACX,uBAAkB,E;;;YAEIB,IAAK,QAAiB,KAAjB,aAAL,EAAY,SAAU,KAAV,a;YACZ,gBAAW,gCA
AwB,KAAxB,C;YACX,yBAAoB,QAAQ,MAAR,I;YACpB,uBAAkB,0BAAwB,WAAU,CAAd,GAAiB,CAAjB,G
AAwB,CAA5C,K;;;QAG1B,iBAAy,C;;IAEpB,C;;MAOiB,Q;MAJb,IAAI,mBAAa,EAAjB,C;QACI,iB;MACJ,IAA
I,mBAAa,CAAjB,C;QACI,MAAM,6B;MACV,aAAa,mE;MAEb,gBAAW,I;MACX,iBAAy,E;MACZ,OAAO,M;I
ACX,C;;MAGI,IAAI,mBAAa,EAAjB,C;QACI,iB;MACJ,OAAO,mBAAa,C;IACxB,C;;;MA/C0C,+D;IAAA,C;;;M
AgEU,0C;QZvgCjD,SYwgCH,sBAAW,kBAAX,EAAuB,YAAvB,EAakD,kBAAID,C;QAAA,OAAwE,KAAK,CA
AT,GAAY,IAAZ,GAAsB,OAAM,CAAN,C;MADtC,C;IAAA,C;;MAHM,0B;QAAA,aAAkB,C;MAAG,0B;QAAA
,aAAsB,K;MAAO,qB;QAAA,QAAa,C;MhBhC7H,IAAI,EgBilCI,SAAS,ChBjlCb,CAAJ,C;QACI,cgBglCkB,oD;Q
hB/kCIB,MAAM,gCAAYB,OAAQ,WAAjC,C;;MgBilCV,OAAO,4BAAwB,SAAxB,EAA8B,UAA9B,EAA0C,KA
A1C,EAAiD,gDAAjD,C;IAGX,C;;MAqB4D,0C;QAAkB,Q;QAAA,oCAAU,sBAAV,EAA0B,YAA1B,EAAqD,kB

AArD,EAAwE,KAAxE,aAAsF,GAAG,UAAH,EAAe,WAAO,OAAtB,CAAtF,O;MAAIB,C;IAAA,C;;MAJc,0B;Q
AAA,aAAkB,C;MAAG,0B;QAAA,aAAsB,K;MAAO,qB;QAAA,QAAa,C;MhBvmCrI,IAAI,EgBwmCI,SAAS,Ch
BxmCb,CAAJ,C;QACI,cgBumCkB,oD;QhBtmCIB,MAAM,gCAAyB,OAAQ,WAAjC,C;;MgBumCV,qBAAGC,O
AAX,UAAW,C;MAEHc,OAAO,4BAAwB,SAAXB,EAA8B,UAA9B,EAA0C,KAA1C,EAAiD,sDAAjD,C;IAEX,C
;;MAiB8E,qB;QAAE,yCAAU,EAAV,C;MAAF,C;IAAA,C;;MADX,0B;QAAA,aAAsB,K;MAAO,qB;QAAA,QAA
a,C;MACzG,OAASe,OAAtE,+BAAkB,UAAIB,UAA2C,UAA3C,EAA+D,KAA/D,CAAsE,EAAI,cAAJ,C;IAAtE,
C;;MAaqD,0B;QAAA,aAAsB,K;MAAO,qB;QAAA,QAAa,C;MAC/F,IAAI,UAAW,OAAx,KAAMb,CAAvB,C;Q
ACI,gBAAGB,WAAW,CAAX,C;QACHB,IAAI,EAAC,SAx8BuC,YAAU,CAw8BID,CAAJ,C;UACI,OAAO,mBA
AM,SAAN,EAAiB,UAAjB,EAA6B,KAA7B,C;;MAI2E,kBAAb,cAAtE,+BAAkB,UAAIB,UAA2C,UAA3C,EAA
+D,KAA/D,CAAsE,C;MV0GtE,kBAAM,iBAaA,qCAAwB,EAAxB,CAAb,C;MAqEA,Q;MAAA,6B;MAAb,OAAa
,cAAb,C;QAAa,sB;QACT,WAAy,WUHLgF,uBVgLiE,IUhLkE,CVgLfH,C;;MUhLhB,OVilo,W;IUHLX,C;;MAU
8E,qB;QAAE,yCAAU,EAAV,C;MAAF,C;IAAA,C;;MADb,0B;QAAA,aAAsB,K;MAAO,qB;QAAA,QAAa,C;MA
CvG,OAASe,OAAtE,6BAAkB,UAAIB,UAA2C,UAA3C,EAA+D,KAA/D,CAAsE,EAAI,mCAAJ,C;IAAtE,C;;MA
SmD,0B;QAAA,aAAsB,K;MAAO,qB;QAAA,QAAa,C;MAC7F,IAAI,UAAW,OAAx,KAAMb,CAAvB,C;QACI,
OAAO,mBAAoB,oBAAd,WAAW,CAAX,CAAc,CAApB,EAAGC,UAAhC,EAA4C,KAA5C,C;;MAG+E,kBAAb,c
AAtE,6BAAkB,UAAIB,UAA2C,UAA3C,EAA+D,KAA/D,CAAsE,C;MViFtE,kBAAM,iBAaA,qCAAwB,EAAxB,
CAAb,C;MAqEA,Q;MAAA,6B;MAAb,OAAa,cAAb,C;QAAa,sB;QACT,WAAy,WUvJgF,uBVuJIE,IUvJkE,CVvJ
hF,C;;MUvJhB,OVwJO,W;IUvJX,C;;MhB7qCI,IAAI,EgBwrCI,SAAS,ChBxrCb,CAAJ,C;QACI,cgBurCkB,oD;Qh
BtrCIB,MAAM,gCAAyB,OAAQ,WAAjC,C;;MgBwrCV,oBAAoB,C;MACpB,gBAAGB,sBAAQ,SAAR,EAAMb,a
AAnB,EAAC,UAAIC,C;MACHB,IAAI,cAAa,EAAb,IAAMb,UAA5C,CAAhC,C;QACI,OAAO,OAAO,SAAK,WA
AZ,C;;MAGX,gBAAGB,QAAQ,C;MACxB,aAAa,iBAAsB,SAAJ,GAAqB,eAAN,KAAM,EAAa,EAAb,CAArB,G
AA2C,EAA7D,C;;QAET,MAAO,WAn4B6E,8BA4B/D,aAn4B+D,EAm4BhD,SAn4BgD,CAAKC,WAm4B/G,C;
QACP,gBAAGB,YAAy,SAAU,OAAtB,I;QAEhB,IAAI,aAAa,MAAO,KAAP,MAAE,QAAQ,CAAR,IAAf,CAAjB,
C;UAA2C,K;QAC3C,YAAy,sBAAQ,SAAR,EAAMb,aAAnB,EAAC,UAAIC,C;;MACP,sBAAa,EAAb,C;MAET
,MAAO,WA14BiF,8BA04BnE,aA14BmE,EA04BpD,gBA14BoD,CAAKC,WA04BnH,C;MACP,OAAO,M;IACX,
C;;MASmD,qB;QAAA,QAAa,C;MAAMb,OAAA,KAAM,eAAM,SAAN,EAAy,KAAZ,C;IAAN,C;;MAOxB,mC
AAGB,MAAhB,EAAwB,IAAxB,EAA8B,IAA9B,E;IAAA,C;;MAOX,OAAe,UAAf,uBAAE,C;IAAf,C;;MkJ/vChD,
0B;MAII,aAC+B,e;MAC/B,cACgC,e;MACHC,WAC6B,e;MAC7B,YAC8B,e;MAC9B,eACiC,e;MACjC,YAC8B,g
B;MAC9B,aAC+B,gB;MAC/B,YAC8B,gB;MAC9B,aAC+B,gB;MAC/B,eACiC,gB;MACjC,iBACmC,gB;MACnC
,sBACwC,gB;MACxC,uBACyC,gB;MACzC,kBACoC,gB;MACpC,cACgC,gB;MACHC,iBACmC,gB;MACnC,iB
ACmC,gB;MACnC,iBACmC,gB;MACnC,YAC8B,gB;MAC9B,aAC+B,iB;MAC/B,aAC+B,iB;MAC/B,uBACyC,i
B;MACzC,wBAC0C,iB;MAC1C,sBACwC,iB;MACxC,uBACyC,iB;MACzC,wBAC0C,iB;MAC1C,sBACwC,iB;
MACxC,cACgC,iB;MACHC,oBACsC,iB;MACtC,cACgC,iB;MACHC,gBACkC,iB;MACiC,aAC+B,iB;MAC/B,mB
ACqC,iB;MACrC,YAC8B,iB;MAC9B,UAC4B,iB;MAC5B,mBACqC,iB;MACrC,gBACkC,iB;MACiC,mBACqC,i
B;MACrC,sBACwC,iB;IAjF5C,C;;;MAAA,iC;QAAA,gB;;MAAA,0B;IAAA,C;;;ICoBA,C;;MAmBmE,6BAAkB
,IAAIB,EAAwB,QAAxB,C;IAAA,C;;MAOC,wBAAM,QAAD,aAAL,C;IAAA,C;;MAS3B,QAAC,iBAaA,a;IAAd,
C;;MAQG,OAAA,iBAaA,a;IAAb,C;;4EAI5C,yB;MAAA,4C;MAAA,mC;QAIyE,MAAM,WAAM,0BAAN,C;MA
AN,C;KAJzE,C;mFAMA,yB;MAAA,4C;MAAA,mC;QAIwE,MAAM,WAAM,0BAAN,C;MAAN,C;KAJxE,C;;M
AQiF,oB;MAAJD,gB;MAAQb,4B;IAAtB,C;;MACW,OAAA,SAAK,aAAL,cAAoB,eAApB,C;IAAA,C;;MAEa,6B
AAkB,SAAlB,EAAwB,4BAaA,QAAb,CAAXB,C;IAAA,C;;MCnEf,gB;IAAD,C;;MAO4F,oB;MAAnG,4B;MAA6
B,oB;MAAsC,sB;IAApE,C;;MACe,OAA2B,aAA1B,YAAM,OAAN,UAAe,gBAAf,CAA0B,EAAW,YAAM,KAAj
B,CAA3B,cAAoD,aAApD,C;IAAA,C;;MACa,2CAAc,gBAAd,EAAyB,YAAzB,EAAgC,0BAAS,QAAT,CAAhC,
C;IAAA,C;;MAGnB,2CAAc,WAAAd,EAA8B,IAAtB,EAA4B,gCAAS,KAAR,C;IAAA,C;;MAUE,gB;IAAD,C;;M
AOgG,oB;MAAvG,4B;MAA+B,oB;MAAwC,sB;IAAxE,C;;MACa,OAA2B,aAA1B,YAAM,OAAN,GAAe,gBAA
W,EAAW,YAAM,KAAjB,CAA3B,cAAoD,aAApD,C;IAAA,C;;MACa,+CAAGB,gBAAhB,EAA2B,YAA3B,EAA
kC,0BAAS,QAAT,CAAI,C;IAAA,C;;MAGnB,+CAAGB,WAAhB,EAAwB,IAAxB,EAA8B,gCAAS,KAAvC,C;I
AAA,C;;MAmBf,oE;MACrB,mB;IAjBJ,C;;MAmBgC,qB;IAAA,C;;MACd,Q;MAFV,YAAy,QAAS,kBAAS,SA
AT,C;MACrB,gBAAsB,uBAAN,KAAM,C;MACZ,IAAI,gDAA+B,4CAAnC,C;QAEN,iBAAiB,mBAAU,SAAV,C;

QACjB,IAAI,mBAAY,SAAZ,gBAAYB,CAAzB,IAA8B,mBAAY,UAAZ,eAAyB,CAA3D,C;UAA8D,gBAAS,QA
AT,C;QAC9D,iB;;;QAGA,mBAAiB,4BAAU,K;QAC3B,IAAI,sDAA+B,kDAAnc,C;UAAgE,gBAAS,QAAT,C;Q
ACrD,8BAAX,YAAW,C;;MATf,qB;IAWJ,C;;MAGI,MAAM,6BAASB,4CAA0C,cAA1C,qCAAqE,QAAR,E,MAAt
B,C;IACV,C;;MCnGmC,6C;IAAA,C;;MA0BnC,gC;MAP8C,0B;IAArB,C;;MAOzB,kC;MACI,YAC4B,aAAS,GA
AT,C;MAE5B,gBACgC,aAAS,wCAAO,kBAAhB,C;IALpC,C;;MASQ,2BAAoB,KAApB,EAA2B,UAA3B,EAAu
C,UAAvC,C;IAAA,C;;;MATR,yC;QAAA,wB;;MAAA,kC;IAAA,C;;MAe6C,oBAAS,CAAC,kBAAV,C;IAAA,C;;
MAGS,oBAAS,qBAAQ,KAAM,cAAvB,C;IAAA,C;;MAGC,oBAAS,qBAAQ,KAAM,cAAvB,C;IAAA,C;;MAGL,
oBAAS,qBAAQ,KAAjB,C;IAAA,C;;MAGG,oBAAS,qBAAQ,KAAjB,C;IAAA,C;;MAGL,oBAAS,qBAAQ,KAAj
B,C;IAAA,C;;MAGG,oBAAS,qBAAQ,KAAjB,C;IAAA,C;;MAGA,OAAA,IAAK,cAAL,GAAa,KAAM,c;IAAnB,
C;;MAGhB,4BAAQ,C;IAAR,C;;MAGA,4BAAQ,C;IAAR,C;;MAGA,OAAM,WAAN,kBAAM,C;IAAN,C;;MAGF
,OAAM,SAAN,kBAAM,C;IAAN,C;;MAGU,OAAI,iBAAJ,GAAMb,IAAD,aAAIB,GAA6B,I;IAA7B,C;;MAEI,O
AAW,iBAAX,IAAK,cAAM,EAAU,KAAM,cAAhB,C;IAAX,C;uHAK/C,yB;MAAA,oC;MAAA,yB;QAaI,cAAc,Y
AAP,WAAO,CAAd,EAAuB,mBAAvB,EAAuC,qBAAvC,EAAyD,qBAAzD,EAA2E,yBAA3E,C;MAAA,C;KAbJ,
C;uHAeA,yB;MAAA,oC;MAAA,yB;QAYI,cAAe,YAAR,YAAQ,CAAF,EAAwB,qBAAxB,EAA0C,qBAA1C,EAA
4D,yBAA5D,C;MAAA,C;KAZJ,C;uHAcA,yB;MAAA,oC;MAAA,yB;QAWI,cAAiB,YAAV,cAAU,CAAjB,EAA0
B,qBAA1B,EAA4C,yBAA5C,C;MAAA,C;KAXJ,C;;MAuBI,cAAiB,uBAAV,cAAU,CAAjB,EAA2B,yBAA3B,C;I
AAA,C;;MAGqC,OAAe,YAAd,eAAU,EAAI,C;IAAf,C;;MAEE,OAAiB,YAAhB,iBAAY,EAAI,C;IAAjB,C;;MAE
A,OAAiB,YAAhB,iBAAY,EAAI,C;IAAjB,C;;MAEI,OAAiB,YAArB,qBAAgB,KAAK,C;IAAtB,C;;MAMG,2BA
AoB,kBAApB,0CAAwC,IAAx,C;IAAA,C;;MAOJ,OAAe,uBAAf,sBAAS,IAAT,CAAe,C;IAAf,C;;MAOF,OAAe
,YAAf,sBAAS,IAAT,CAAe,C;IAAf,C;;MAGV,6D;IAAA,C;;MAGC,8D;IAAA,C;;MAGE,gE;IAAA,C;;MAGA,gE
;IAAA,C;;MAGK,qE;IAAA,C;;MAGA,qE;IAAA,C;;MAGD,oE;IAAA,C;;MAWF,kE;IAAA,C;;MASC,mE;IAAA,
C;;MasBrB,UAAx,M;MAnBJ,sB;QAD8B,OAcD,kBAAM,W;WAcIb,2BAAS,GAAT,C;QAF8B,OAEd,I;;QAEZ,
YAAy,kBAAc,c;QAC1B,qBAAiB,KAAjB,C;QACA,sBAAkB,CAAIB,C;QAEI,YAAQ,MAAR,C;UAAqC,kD;UA
AO,eAAa,I;UAApB,OjK7I1C,S;;ciK8IK,YAAQ,CAAR,C;UAAcS,wD;UAAO,gBAAc,C;UAArB,OjK9I3C,W;;ci
K+IK,YAAQ,MAAR,C;;aACA,YAAQ,SAAR,C;;aACA,YAAQ,KAAR,C;;aACA,YAAQ,MAAR,C;;aACA,YAAQ
,MAAR,C;;aACA,YAAQ,MAAR,C;;aACA,YAAQ,UAAU,KAAIB,C;;UAC0B,iD;UAAO,eAAa,I;UAApB,OjKtJ/
B,W;;QiK4IC,e;QAYA,YAAy,sBAAS,IAAT,C;QAER,iB;UAAc,0BAAiB,KAAjB,C;aAcD,oBAAc,CAAd,C;UA
AmB,4BAAMb,KAAAnB,EAA0B,aAA1B,C;;UACX,+BAASB,KAAAtB,EAA6B,iBvE+FH,MAAW,KuE/FM,KvE+
N,CuE/FR,CAA7B,C;;QAvBc,OAoB1B,SAIS,UAAL,IAAK,C;;IAxBiB,C;;MA6B9B,YAAQ,CAAR,C;QADwC,O
AC3B,C;WAcB,YAAQ,EAAR,C;QAFwC,OAElB,C;WAcD,YAAQ,GAAR,C;QAHwC,OAGzB,C;;QAHyB,OAIh
C,C;IAJgC,C;;MAoBJ,wB;QAAA,WAAgB,C;MAI7C,Q;MrKxPX,IAAI,EqKqPQ,YAAy,CrKrPpB,CAAJ,C;QACI
,cqKoPyB,oD;QrKnPzB,MAAM,gCAAyB,OAAQ,WAAjC,C;;MqKoPN,IAAI,iBAAJ,C;QAAkB,OAAO,kBAAM,
W;MAC/B,aAAa,sBAAS,IAAT,C;MAET,IvEiEsC,MAAW,KuEjE7C,MvEiE6C,CuEjEjD,GAAc,MAAd,C;QAAs
B,6BAASB,MAAtB,EAAuC,eAAT,QAAS,EAAa,EAAb,CAAvC,C;;QACd,wBAAiB,MAAjB,C;MAFZ,OAAO,O
AGE,UAAL,IAAK,C;IACb,C;;MnF/QgB,gBAAhB,sB;MmFiSI,IAAI,iBAAJ,C;QAAkB,yBAAO,EAAP,C;MACIB
,yBAAO,IAAP,C;MACc,YAAd,kB;MA1LA,YAAe,YAAR,aAAQ,C;MAAf,cAAwB,sB;MAAxB,cAA0C,sB;MAA
1C,kBAA4D,0B;MA2LxD,eAAe,UAAS,C;MACxB,iBAAiB,YAAW,CAAX,IAAgB,gBA Ae,C;MACHD,iBAAiB,
YAAW,CAAX,KAAiB,cAAc,QAA/B,C;MACjB,IAAI,QA AJ,C;QACI,yBAAO,KAAP,CAAc,gBAAO,EAAP,C;;
MAEIB,IAAI,UAAJ,C;QACI,yBAAO,OAAP,CAAgB,gBAAO,EAAP,C;;MAEpB,IAAI,eAAe,CAAC,QAAD,IAA
a,CAAC,UAA7B,CAAJ,C;QACI,yBAAO,OAAP,C;QACA,IAAI,gBAAe,CAAnB,C;UACI,yBAAO,EAAP,C;UAC
A,UAAiC,WAAvB,WAAy,WAAW,EAAS,CAAT,EAAY,EA AZ,C;UAE7B,kBAAc,OAAd,KAA2B,CAA3B,C;Y
AAgC,yBAAO,GAAP,EAAY,CAAZ,EAAe,CAAF,C;eAChC,kBAAc,IAAd,KAAuB,CAAvB,C;YAA4B,yBAAO,
GAAP,EAAY,CAAZ,EAAe,CAAF,C;;YACpB,yBAAO,GAAP,C;;QAGhB,yBAAO,EAAP,C;;MAxBuB,OjK7N5B,
S8EnE8B,W;ImFgSF,C;;;MApSvC,c;MAiBkD,8D;MAjBID,a;IAAA,C;;MAAA,2IAiBkD,sDAjBID,G;IAAA,C;;
MAyU0D,OA AW,aAAX,SAAW,EA AW,IAAX,C;IAAX,C;;MAKC,OA AW,aAAX,oBAAW,EA AW,IAAX,C;IAA
X,C;;MAKE,oBAAS,oBAAoB,SAAPB,EAA0B,IAA1B,yCAAT,C;IAAA,C;;MAQ1B,oE;IAAA,C;;MAKC,sE;IAA
A,C;;MAKE,sE;IAAA,C;;MAKF,qE;IAAA,C;;MAKC,uE;IAAA,C;;MAKE,uE;IAAA,C;;MAKH,qE;IAAA,C;;MA
KC,uE;IAAA,C;;MAKE,uE;IAAA,C;;MAKR,gE;IAAA,C;;MAKC,kE;IAAA,C;;MAKE,kE;IAAA,C;;MAKH,gE;I

AAA,C;;MAKC,ke;IAAA,C;;MAKE,ke;IAAA,C;;MAKL,8D;IAAA,C;;MAKC,gE;IAAA,C;;MAKE,gE;IAAA,C;;
MAKJ,6D;IAAA,C;;MAKC,+D;IAAA,C;;MAKE,+D;IAAA,C;;MAOsC,8BAAW,SAAX,C;IAAA,C;;MAMG,8BA
AW,SAAX,C;IAAA,C;;MC/ZxB,QAAM,SAAN,M;aAC5C,a;UAD4C,OACHb,I;aAC5B,c;UAF4C,OAef,I;aAC7B,
c;UAH4C,OAGf,I;aAC7B,S;UAJ4C,OAIpB,G;aACxB,S;UAL4C,OAKpB,G;aACxB,O;UAN4C,OAMtB,G;aAcTb,
M;UAP4C,OAovB,G;gBAPuB,mC;;IAAA,C;;IChDhD,C;;qFCAA,yB;MAAA,wB;MAAA,wB;QA2BI,WAhBO,c
AgBI,U;QAhBkB,KAIb7B,E;QAJBA,OAKBO,IAAK,a;MAjBhB,C;KAZA,C;;MA2BI,WAAW,mB;MACX,O;MA
CA,OAAO,IAAK,a;IACHb,C;;MAWgC,kB;MAAc,wB;IAAf,C;;MAAC,iB;IARhC,C;;MAQ8C,oB;IAR9C,C;;MA
AA,sBAQgC,qCARhC,EAQ8C,8CAR9C,C;IAAA,C;;MAAA,OAQgC,iDARhC,IAQ8C,8CAR9C,O;IAAA,C;;MA
AA,c;MAQgC,sD;MAAc,yD;MAR9C,a;IAAA,C;;MAAA,4IAQgC,sCARhC,IAQ8C,4CAR9C,I;IAAA,C;iGAUA,y
B;MAAA,wB;MAGBA,8C;MAhBA,wB;QA6BI,WAhBO,cAgBI,U;QACX,aAjBmC,KAIbTb,E;QAJBb,OAKBO,oB
AAW,MAAX,EAAMb,IAAK,aAAXB,C;MAjBX,C;KAdA,C;mGAgBA,yB;MAAA,8C;MAAA,mC;QAaI,WAAW,
mB;QACX,aAAa,O;QACb,OAAO,oBAAW,MAAX,EAAMb,IAAK,aAAXB,C;MACX,C;KAhBA,C;4EC1DA,yB;
MAAA,8B;MAAA,4B;QAOyC,Q;QAAA,gFAAoB,C;MAApB,C;KAPzC,C;;MCqDI,qC;MA/CuB,kB;MAAgB,kB;
MAAgB,kB;MAMvD,iBAAsB,iBAAU,UAAV,EAaiB,UAAjB,EAawB,UAAxB,C;IANA,C;;M1KmBtB,IAAI,E0
KViB,CAAT,sBAAY,GAAZ,KAA4C,CAAT,sBAAY,GAA/C,MAA+E,CAAT,sBAAY,GAAIF,C1KUR,CAAJ,C;
QACI,c0KVI,2E;Q1KWJ,MAAM,gCAAYB,OAAQ,WAAjC,C;;M0KTN,OAAO,CAAA,KAAM,IAAI,EAAV,KAA
gB,KAAM,IAAI,CAA1B,IAA+B,KAA/B,I;IACX,C;;MAKkC,OAAE,UAAF,oBAAS,UAAT,SAAGB,U;IAAhB,C;;
MAIV,gB;MADpB,IAAI,SAAS,KAAb,C;QAAoB,OAAO,I;MACP,iE;MAAD,mB;QAA6B,OAAO,K;;MAAvD,m
BAAMb,M;MACnB,OAAO,IAAK,UAAI,KAAgB,YAAa,U;IACxC,C;;MAE+B,qB;IAAA,C;;MAEqB,wBAAU,K
AAM,UAAhB,I;IAAA,C;;MAOhD,OAAA,IAAK,MAAL,GAAa,KAAb,KAAuB,IAAK,MAAL,KAAc,KAAAd,IACf
,IAAK,MAAL,IAAc,KADtB,C;IAAA,C;;MAQA,OAAA,IAAK,MAAL,GAAa,KAAb,KAAuB,IAAK,MAAL,KAA
c,KAAAd,KACd,IAAK,MAAL,GAAa,KAAb,KAAsB,IAAK,MAAL,KAAc,KAAAd,IACf,IAAK,MAAL,IAAc,KADr
B,CADc,CAAvB,C;IAAA,C;;MAIJ,uC;MACI,2BAIuC,G;MAEvC,eAIoC,kBAAC,CAAd,EAaiB,CAAjB,EAoB,
EAApB,C;IAXxC,C;;MAAA,8C;QAAA,6B;;MAAA,uC;IAAA,C;;MA9CA,uD;MAG6C,0BAAK,KAAL,EAAY,
KAAZ,EAAMb,CAAnB,C;MAH7C,Y;IAAA,C;4FCKA,yB;MAAA,2D;MAAA,4B;QAAQ,MAAM,6BAAoB,6BA
ApB,C;MAAN,C;KAAR,C;;MCYuC,+BAAoB,KAApB,C;IAAA,C;;MASiD,sB;IAAA,C;;MAE5F,e;MAAA,iB;
MAAA,uB;IAAA,C;;MAAA,8C;MAAA,C;MAKI,wF;MAKA,sF;MAMA,wE;IAhBJ,C;;MAKI,iC;MAAA,iD;IAA
A,C;;MAKA,iC;MAAA,gD;IAAA,C;;MAMA,iC;MAAA,yC;IAAA,C;;MAhBJ,iJ;IAAA,C;;MAAA,a;aAAA,c;U
AAA,sD;aAAA,a;UAAA,qD;aAAA,M;UAAA,8C;gBAAA,gE;IAAA,C;;MAyBA,mC;IAAA,C;;MAAA,0C;QA
AA,yB;;MAAA,mC;IAAA,C;;MAII,qBAAsC,W;MACtC,gBAA2B,iC;IAFK,C;;MAWjB,Q;MALP,IAAI,kBAAW,
iCAAf,C;QACI,gBAAS,mC;QACT,qBAAc,I;;MAGiB,OAAO,gF;IACX,C;;MAEoC,yBAAW,iC;IAAX,C;;MAEN,
OAAI,oBAAJ,GAA2B,SAAN,UAAAM,CAA3B,GAA2C,iC;IAA3C,C;;MAEA,+BAAoB,UAApB,C;IAAA,C;;MA
GI,4B;IAAD,C;;MAAC,2B;IAAA,C;;MAEE,W;IAAA,C;;MAEN,OAAM,SAAN,UAAAM,C;IAAN,C;;oFC2CtC,yB
;MAAA,gD;MAAA,4B;QAM6C,OAAmB,aAAIB,YAAY,GAAM,C;MAAnB,C;KAN7C,C;oGAQA,yB;M9E7FA,i
B;M8E6FA,4B;QAMqD,O9E5FqB,a8E4FpB,YAAY,G9E5FQ,C8E4FrB,GAA6C,EAA7C,I;MAAA,C;KANrD,C;s
GAQA,yB;MAAA,ke;MAAA,4B;QAMsD,OAAmB,sBAAIB,YAAW,GAAO,C;MAAnB,C;KANtD,C;8FAQA,yB;
MAAA,0D;MAAA,0B;MAAA,4B;QAOmD,OAAuC,OAApB,kBAAIB,YAAY,GAAM,CAAoB,C;MAAvC,C;KA
PnD,C;4FASA,yB;MAAA,wD;MAAA,0B;MAAA,4B;QAOkD,OAA2B,OAAAnB,iBAAR,SAAQ,CAAMb,C;MAA
3B,C;KAPID,C;;MAuBI,OAA+E,OAA9E,SAAQ,KAAI,WAAa,CAAjB,CAAR,GAAkD,CAAIB,YAAY,GAAM,M
AAK,CAAL,IAAU,WAAa,CAAvB,CAA4B,C;IAA/E,C;;MAeA,OAA+E,OAA9E,SAAQ,IAAI,CAAJ,IAAS,WAA
a,CAAtB,CAAR,GAAwD,CAAIB,YAAY,GAAM,OAAK,WAAa,CAAIB,CAAsB,C;IAA/E,C;oFAEJ,yB;MAAA,g
D;MAAA,4B;QAM8C,OAAqB,aAApB,YAAY,KAAQ,C;MAArB,C;KAN9C,C;oGAQA,yB;M9EtKA,iB;M8EsKA
,4B;QAOI,O9EtKsE,a8EsKrE,YAAY,K9EtKyD,C8EsKtE,GAA+C,EAA/C,I;MAAA,C;KAPJ,C;sGASA,yB;MAA
A,ke;MAAA,4B;QAMuD,OAAqB,sBAApB,YAAW,KAAS,C;MAArB,C;KANvD,C;8FAQA,yB;MAAA,0D;MA
AA,4B;MAAA,4B;QAOqD,OAAyC,QAAPb,kBAApB,YAAY,KAAQ,CAAoB,C;MAAzC,C;KAPrD,C;4FASA,y
B;MAAA,wD;MAAA,4B;MAAA,4B;QAOoD,OAA2B,QAAnB,iBAAR,SAAQ,CAAMb,C;MAA3B,C;KAPpD,C;;
MAuBI,OAAoF,QAAnF,SAAQ,KAAI,WAAa,EAajB,CAAR,GAAqD,CAApB,YAAY,KAAQ,MAAK,EAAL,IA
AW,WAAa,EAAXB,CAA8B,C;IAApF,C;;MAeA,OAAoF,QAAnF,SAAQ,IAAI,EAajB,IAAU,WAAa,EAavB,CAA

R,GAA4D,CAApB,YAA Y,KAAQ,OAAK,WAAa,EAAIB,CAAuB,C;IAApF,C;0E7KIRJ,yB;MAaA,kF;MAbA,wB ;QAUbI,IAAI,CAbI,KAAr,C;UACI,cAda,qB;UAeb,MAAM,8BAAyB,OAAQ,WAAjC,C;;MAdd,C;KAXA,C;0EA aA,yB;MAAA,kF;MAAA,qC;QAUl,IAAI,CAAC,KAAL,C;UACI,cAAc,a;UACd,MAAM,8BAAyB,OAAQ,WAAj C,C;;MAEd,C;KAdA,C;sFAGbA,yB;MAWA,kF;MAXA,wB;QAQW,yB;QAEp,IAfSb,KAeIB,QAAJ,C;UACI,cAh B2B,0B;UAIb3B,MAAM,8BAAyB,OAAQ,WAAjC,C;;UAEN,wBAnBkB,K;;QAAtB,4B;MACJ,C;KATA,C;wFA WA,yB;MAAA,kF;MAAA,qC;QAYI,IAAI,aAAJ,C;UACI,cAAc,a;UACd,MAAM,8BAAyB,OAAQ,WAAjC,C;;U AEN,OAAO,K;;MAEf,C;KAIBA,C;oEAoBA,yB;MAaA,4E;MAbA,wB;QAUbI,IAAI,CAbE,KAAr,C;UACI,cAdW ,e;UAeX,MAAM,2BAAsB,OAAQ,WAA9B,C;;MAdd,C;KAXA,C;sEAaA,yB;MAAA,4E;MAAA,qC;QAUl,IAAI, CAAC,KAAL,C;UACI,cAAc,a;UACd,MAAM,2BAAsB,OAAQ,WAA9B,C;;MAEd,C;KAdA,C;kFAGbA,yB;MAc A,4E;MAdA,wB;QAWW,uB;QAEp,IAfoB,KAehB,QAAJ,C;UACI,cAhByB,0B;UAIbZb,MAAM,2BAAsB,OAAQ ,WAA9B,C;;UAEN,sBAnBgB,K;;QAApB,0B;MACJ,C;KAZA,C;oFAcA,yB;MAAA,4E;MAAA,qC;QAYI,IAAI,a AAJ,C;UACI,cAAc,a;UACd,MAAM,2BAAsB,OAAQ,WAA9B,C;;UAEN,OAAO,K;;MAEf,C;KAIBA,C;oEAqB A,yB;MAAA,4E;MAAA,0B;QAMiD,MAAM,2BAAsB,OAAQ,WAA9B,C;MAAN,C;KANjD,C;;M6C1DI,8B;MA lDA,kB;IAD8B,C;;MAUQ,iD;IAAA,C;;MAMA,gD;IAAA,C;wFAItC,yB;MAAA,gB;MAAA,8B;MAAA,mB;QA WgB,Q;QADR,mB;UADJ,OACiB,I;;UADjB,OAeY,2E;MAFZ,C;KATJ,C;;MASBQ,kBADE,UACF,kB;QADJ,OA CkB,UAAM,U;;QADxB,OAeY,I;IAFZ,C;;MAYI,kBADE,UACF,kB;QADJ,OACkB,UAAM,W;;QADxB,OAeY,s BAAU,UAAV,O;IAFZ,C;;MAOJ,gC;IAAA,C;wHAKI,yB;MAAA,iC;MAAA,wB;QAKI,uBAAO,KAAP,C;MAAA ,C;KALJ,C;wHAOA,yB;MAAA,kD;MAAA,iC;MAAA,4B;QAKI,uBAAO,cAAc,SAAd,CAAP,C;MAAA,C;KALJ, C;;MAZJ,uC;QAAA,sB;;MAAA,gC;IAAA,C;;MAqBI,0B;IADkB,C;;MAI0B,+CAAoB,uBAAa,KAAM,UAAAnB, C;IAApB,C;;MACb,OAAU,SAAV,cAAU,C;IAAV,C;;MACG,oBAAU,cAAV,M;IAAA,C;;MA3F1C,c;MAOI,s D;MAPJ,a;IAAA,C;;MAAA,2IAOI,sCAPJ,G;IAAA,C;;MASGI,OAAO,mBAAQ,SAAR,C;IAAP,C;;MAUA,IAAI,8 CAAJ,C;QAA6B,MAAM,eAAM,U;IAC7C,C;gFAEA,yB;MAAA,4B;MAAA,qB;MAtCQ,kD;MASCR,wB;QAOW ,Q;;UACI,OAhdH,WAgDW,OAhdX,C;;UAI DN,gC;YACS,OA3CH,WAAO,cA2CI,CA3CJ,CAAP,C;;YAwCD, O;;QAAP,W;MAKJ,C;KAZA,C;kFAcA,yB;MAAA,4B;MAAA,qB;MAPDQ,kD;MAoDR,mC;QAOW,Q;;UACI,O A9DH,WA8DW,gBA9DX,C;;UA+DN,gC;YACS,OAzDH,WAAO,cAyDI,CAzDJ,CAAP,C;;YAsDD,O;;QAAP, W;MAKJ,C;KAZA,C;8EAgBA,yB;MAAA,oD;MAAA,gB;MAAA,8B;MAAA,4B;QAUW,Q;QADP,yB;QACA,O AAO,gF;MACX,C;KAXA,C;+EAaA,yB;MAAA,gB;MAAA,8B;MAAA,uC;QAegB,UADL,M;QAAM,gBAAgB,2 B;QACzB,sB;UAAQ,yF;;UACA,mBAAU,SAAV,C;QAFZ,a;MAIJ,C;KAIBA,C;kFAoBA,yB;MAAA,gB;MAAA,8 B;MAAA,0C;QAUW,Q;QADP,IAAI,mBAAJ,C;UAAe,OAAO,Y;QACtB,OAAO,gF;MACX,C;KAXA,C;qEAaA,y B;MAAA,gB;MAAA,8B;MAAA,kD;QAIb0B,UADf,M;QAAM,gBAAgB,2B;QACzB,sB;UAAQ,mBAAU,gFAAV ,C;;UACA,mBAAU,SAAV,C;QAFZ,a;MAIJ,C;KApBA,C;mEAwBA,yB;MAAA,4B;MAAA,gB;MAAA,8B;MAA A,uC;YAe8C,I;YADnC,M;QACH,wB;UAAa,gB;UAAO,SA3JhB,WA2JwB,UAAU,gFAAV,CA3JxB,C;;UA4JI,o BAAO,eAAP,C;QAFZ,a;MAIJ,C;KAIBA,C;gFAoBA,yB;MAAA,gB;MAAA,8B;MAAA,iC;MA1GA,qB;MAPDQ, kD;MA8JR,uC;QAWW,Q;QACH,wB;UA/GG,U;;YA+GkC,U;YA9G9B,SA9DH,gBA4KuB,UAAU,sFAAV,CA5K vB,C;;YA+DN,gC;cACS,SAzDH,gBAAO,cAyDI,CAzDJ,CAAP,C;;cAsDD,O;;UA+GU,a;;UACL,uBAAO,eAAP ,C;QAFZ,W;MAIJ,C;KafA,C;wEAIbA,yB;MAAA,4B;MAAA,uC;QAcW,Q;QAAM,gBAAgB,2B;QACzB,sB;UA AQ,gB;;UACO,OAjMX,WAI MmB,UAAU,SAAV,CAjMnB,C;;QA+LR,W;MAIJ,C;KAIBA,C;wFAoBA,yB;MA/ lA,4B;MAAA,qB;MAPDQ,kD;MAmMR,uC;QAYW,Q;QADP,YAA Y,e;QACC,gBAAgB,2B;QACzB,sB;UAAQ,g B;;UARJL,U;;YACI,SA9DH,WAmNkB,oBAnNIB,C;;YA+DN,gC;cACS,SAzDH,WAAO,cAyDI,CAzDJ,CAAP,C; ;cAsDD,O;;UAsJK,a;;QAFZ,W;MAIJ,C;KAhBA,C;;MA8BI,Q;MAAA,iD;QAAyB,Y;;MACzB,OAAO,S;IACX,C; 4EAEA,yB;MAAA,gB;MAAA,8B;MAAA,oC;QAU0B,Q;QAAtB,IAAI,mBAAJ,C;UAAe,OAAO,gFAAP,C;;QAC f,OAAO,S;MACX,C;KAZA,C;;MzCjTiC,uB;QAAA,UAAkB,kC;mBAA4C,O;;IAA/D,C;;0DAEHc,yB;MAAA,2D; MAAA,mB;QAKoC,MAAM,8B;MAAN,C;KALpC,C;oEAOA,yB;MAAA,2D;MAAA,yB;QAMkD,MAAM,6BAA oB,sCAAmC,MAAvD,C;MAAN,C;KANID,C;;MAoBI,OAAO,O;IACX,C;;MAYI,OAAO,gB;IACX,C;;MAYI,OA AgB,MAAT,QAAS,C;IACpB,C;;MAYI,gB;MACA,OAAO,S;IACX,C;;MAaI,MAAM,SAAN,C;MACA,OAAO,S;I ACX,C;;MAYI,OAAO,MAAM,SAAN,C;IACX,C;;MAWI,OAAW,UAAU,SAAV,CAAJ,GAAqB,SAArB,GAA+B, l;IAC1C,C;;MAWI,OAAW,CAAC,UAAU,SAAV,CAAL,GAAsB,SAAtB,GAAgC,I;IAC3C,C;;MAaI,iBAAc,CAA d,UAAsB,KAAtB,U;QACI,OAAO,KAAP,C;;IAER,C;;M0K3IkF,Y;IAAA,C;;MCa9E,kB;MACA,oB;IAFgC,C;;M

AQS,aAAG,UAAH,UAAW,WAAX,M;IAAA,C;;;MAPzC,iB;IAhBJ,C;;MAiBI,kB;IAjBJ,C;;MAAA,gBAgBI,qCAhBJ,EAiBI,wCAjBJ,C;IAAA,C;;MAAA,c;MAgBI,sD;MACA,uD;MAjBJ,a;IAAA,C;;MAAA,4IAgBI,sCAhBJ,IAiBI,wCAjBJ,I;IAAA,C;;MAgCoD,gBAAK,SAAL,EA AW,IAAX,C;IAAA,C;;MAMN,iBAAO,eAAP,EAAC,gBAAd,E;IAAA,C;;MAkBI,C,kB;MACA,oB;MACA,kB;IAHyC,C;;MASA,aAAG,UAAH,UAAW,WAAX,UAAoB,UAApB,M;IAAA,C;;;MARzC,iB;IAhBJ,C;;MAiBI,kB;IAjBJ,C;;MAkBI,iB;IAIBJ,C;;MAAA,kBAgBI,qCAhBJ,EAiBI,wCAjBJ,EAkBI,qCAIBJ,C;IAAA,C;;MAAA,c;MAgBI,sD;MACA,uD;MACA,sD;MAIBJ,a;IAAA,C;;MAAA,4IAgBI,sCAhBJ,IAiBI,wCAjBJ,IAkBI,sCAIBJ,I;IAAA,C;;MA+BmD,iBAAO,eAAP,EAAC,gBAAd,EAAsB,eAAtB,E;IAAA,C;;ICrEnD,C;;;M3LII,6B;MAFyD,gB;IAAnC,C;;MAEtB,+B;MACI,iBAGoC,UAAM,CAAN,C;MAEpC,iBAGoC,UAAM,MAAN,C;MAEpC,kBAGmC,C;MAEnC,iBAGkC,C;IANtC,C;;;MAAA,sC;QAAA,qB;;MAAA,+B;IAAA,C;;MA6BmE,OAAa,0BAgJ1C,SAAL,GAAiB,GAhJ8B,EAAU,KAgJpD,KAAL,GAAiB,GAhJ8B,C;IAAb,C;;MAQR,OAAa,0BAwIIC,SAAL,GAAiB,GAxIsB,EAAU,KEyI5C,KAAL,GAAiB,KFzIsB,C;IAAb,C;sGAE3D,yB;MA+JA,6B;MChKA,8C;MDCA,wB;QAMyD,OCAS,YAAiB,CDkKhD,cAAU,SAAL,GAAiB,GAAtB,CCIKgD,MAAjB,EDAe,KCAc,KAA7B,C;MDAT,C;KANzD,C;sGAQA,yB;MAiKA,WAS6D,wB;MAT7D,+B;MgClKA,gD;MhCCA,wB;QAM0D,OgCAS,aAAkB,ChCoKhD,eAAW,oBAAL,SAAK,CAAL,UAAN,CgCpKgD,MAAIB,EhCAgB,KgCAc,KAA9B,C;MhCAT,C;KAN1D,C;4FAQA,yB;MA+IA,6B;MA/IA,wB;QAEsD,OCMD,cAAU,CDgJ5B,cAAU,SAAL,GAAiB,GAAtB,CChJ4B,MAAK,GA AW,CDgJ5C,cAtJsC,KAsJ5B,KAAL,GAAiB,GAAtB,CChJ4C,MAAX,IAAf,C;MDNC,C;KAFtD,C;4FAGA,yB;MA4IA,6B;MA5IA,wB;QAEuD,OCGF,cAAU,CDgJ5B,cAAU,SAAL,GAAiB,GAAtB,CChJ4B,MAAK,GA AW,CCiJ5C,cFpJuC,KEoJ7B,KAAL,GAAiB,KAAtB,CDjJ4C,MAAX,IAAf,C;MDHE,C;KAFvD,C;4FAGA,yB;MAyIA,6B;MAzIA,wB;QAEqD,OCAA,cAAU,CDgJ5B,cAAU,SAAL,GAAiB,GAAtB,CChJ4B,MAAK,GDAI,KCAO,KAAX,IAAf,C;MDAA,C;KAFrD,C;4FAGA,yB;MAgJA,WAS6D,wB;MAT7D,+B;MAhJA,wB;QAEuD,OgCAA,eAAW,ChCuJ7B,eAAW,oBAAL,SAAK,CAAL,UAAN,CgCvJ6B,MAAK,KhCAI,KgCAO,KAAX,CAAhB,C;MhCAA,C;KAFvD,C;8FAIA,yB;MAiIA,wB;QAEuD,OCMD,cAAU,CDmI7B,cAAU,SAAL,GAAiB,GAAtB,CCnI6B,MAAK,GAAY,CDmI9C,cAzIwC,KAyI9B,KAAL,GAAiB,GAAtB,CCnI8C,MAAZ,IAAf,C;MDNC,C;KAFvD,C;8FAGA,yB;MA+HA,6B;MA/HA,wB;QAEwD,OCGF,cAAU,CDmI7B,cAAU,SAAL,GAAiB,GAAtB,CCnI6B,MAAK,GAAY,CCoI9C,cFvIyC,KEuI/B,KAAL,GAAiB,KAAtB,CDpI8C,MAAZ,IAAf,C;MDHE,C;KAFxD,C;8FAGA,yB;MA4HA,6B;MA5HA,wB;QAEsD,OCAA,cAAU,CDmI7B,cAAU,SAAL,GAAiB,GAAtB,CCnI6B,MAAK,GDAK,KCAO,KAAZ,IAAf,C;MDAA,C;KAFtD,C;8FAGA,yB;MAmIA,WAS6D,wB;MAT7D,+B;MANIA,wB;QAEwD,OgCAA,eAAW,ChC0I9B,eAAW,oBAAL,SAAK,CAAL,UAAN,CgC1I8B,MAAK,UhCAK,KgCAO,KAAZ,CAAhB,C;MhCAA,C;KAFxD,C;8FAIA,yB;MAqHA,6B;MArHA,wB;QAEuD,OCMD,cAAe,YAAL,CDsH7B,cAAU,SAAL,GAAiB,GAAtB,CCTH6B,MAAK,EAAY,CDsH9C,cA5HwC,KA4H9B,KAAL,GAAiB,GAAtB,CCTH8C,MAAZ,CAAf,C;MDNC,C;KAFvD,C;8FAGA,yB;MAkHA,6B;MAIHA,wB;QAEwD,OCGF,cAAe,YAAL,CDsH7B,cAAU,SAAL,GAAiB,GAAtB,CCTH6B,MAAK,EAAY,CCuH9C,cFIHyC,KE0H/B,KAAL,GAAiB,KAAtB,CDvH8C,MAAZ,CAAf,C;MDHE,C;KAFxD,C;8FAGA,yB;MA+GA,6B;MA/GA,wB;QAEsD,OCAA,cAAe,YAAL,CDsH7B,cAAU,SAAL,GAAiB,GAAtB,CCTH6B,MAAK,EDAK,KCAO,KAAZ,CAAf,C;MDAA,C;KAFtD,C;8FAGA,yB;MAshA,WAS6D,wB;MAT7D,+B;MAthA,wB;QAEwD,OgCAA,eAAW,ChC6H9B,eAAW,oBAAL,SAAK,CAAL,UAAN,CgC7H8B,MAAK,UhCAK,KgCAO,KAAZ,CAAhB,C;MhCAA,C;KAFxD,C;0FAIA,yB;MAwGA,6B;MCIGA,4C;MDNA,wB;QAEqD,OCMD,WdyGjB,cAAU,SAAL,GAAiB,GAAtB,CCzGiB,EDyGjB,cA/GoC,KA+G1B,KAAL,GAAiB,GAAtB,CCzGiB,C;MDNC,C;KAFrD,C;0FAGA,yB;MAqGA,6B;MCIGA,4C;MDHA,wB;QAEsD,OCGF,WdyGjB,cAAU,SAAL,GAAiB,GAAtB,CCzGiB,EC0GjB,cF7GqC,KE6G3B,KAAL,GAAiB,KAAtB,CD1GiB,C;MDHE,C;KAFtD,C;0FAGA,yB;MAkGA,6B;MCIGA,4C;MDAA,wB;QAEoD,OCAA,WdyGjB,cAAU,SAAL,GAAiB,GAAtB,CCzGiB,EDAkB,KCAIB,C;MDAA,C;KAFpD,C;0FAGA,yB;MAyGA,WAS6D,wB;MAT7D,+B;MgCzGA,8C;MhCAA,wB;QAEsD,OgCAA,YhCgHjB,eAAW,oBAAL,SAAK,CAAL,UAAN,CgChHiB,EhCAmB,KgCANB,C;MhCAA,C;KAFtD,C;0FAIA,yB;MA2FA,6B;McrFA,kD;MDNA,wB;QAEqD,OCMD,cD4FjB,cAAU,SAAL,GAAiB,GAAtB,CC5FiB,ED4FjB,cAlGoC,KAKG1B,KAAL,GAAiB,GAAtB,CC5FiB,C;MDNC,C;KAFrD,C;0FAGA,yB;MAwFA,6B;McrFA,kD;MDHA,wB;QAEsD,OCGF,cD4FjB,cAAU,SAAL,GAAiB,GAAtB,CC5FiB,EC6FjB,cFhGqC,KEgG3B,KAAL,GAAiB,KAAtB,CD7FiB,C;MDHE,C;KAFtD,C;0FAGA,yB;MAqFA,6B;McrFA,kD;MDAA,wB;QAEoD,OCAA,cD4FjB,cAAU,SAAL,GAAiB,GAAtB,CC5FiB,EDAkB,KCAIB,C;MDAA,C;KAFpD,C;0FAGA,yB;MA4FA,WAS6D,wB;MAT7D,+B;

MgC5FA,oD;MhCAA,wB;QAEsD,OgCAA,ehCmGjB,eAAW,oBAAL,SAAK,CAAL,UAAN,CgCnGiB,EhCAmB,KgCAnB,C;MhCAA,C;KAFtD,C;0EAIa,yB;MAAA,0B;MAAA,+B;MAAA,mB;QAE0C,sBAAW,OAAL,SAAK,KAAX,C;MAAA,C;KAF1C,C;0EAGA,yB;MAAA,0B;MAAA,+B;MAAA,mB;QAE0C,sBAAW,OAAL,SAAK,KAAX,C;MAAA,C;KAF1C,C;kGAIA,yB;MAAA,8C;MAUEA,6B;MAvEA,wB;QAE8D,0BA8E3B,cAAU,SAAL,GAAiB,GAAtB,CA9E2B,EA8E3B,cA9EoD,KA8E1C,KAAL,GAAiB,GAAtB,CA9E2B,C;MAAA,C;KAF9D,C;0FAIA,yB;MAAA,+B;MgK5IJ,0B;MhK4II,wB;QAEmD,sBgK3IgC,OhK2I1B,IAAK,KgK3IX,GhK2IoB,KAAM,KgK3IM,ChK2IhC,C;MAAA,C;KAFnD,C;wFAGA,yB;MAAA,+B;MgK1IJ,0B;MhK0II,wB;QAEkD,sBgKzI+B,OhKyIzB,IAAK,KgKzIX,GhKyImB,KAAM,KgKzIM,ChKyI/B,C;MAAA,C;KAFID,C;0FAGA,yB;MAAA,+B;MgKxIJ,0B;MhKwII,wB;QAEmD,sBgKvIgC,OhKuI1B,IAAK,KgKvIX,GhKuIoB,KAAM,KgKvIM,ChKuIhC,C;MAAA,C;KAFnD,C;0EAGA,yB;MAAA,+B;MgKtIJ,0B;MhKsII,mB;QAEiC,sBgKrlqB,OAAP,ChKqIR,SgKrIe,ChKqIrB,C;MAAA,C;KAFjC,C;;MAAmC,gB;IAAA,C;kFACnC,yB;MgK9IJ,4B;MhK8II,mB;QASqC,OgKpJiD,QhKoJ5C,SgKpJY,GhKoJE,GgKpJ8B,C;MhKoJjD,C;KATrC,C;;MAmBiC,OAAK,SAAL,GAAiB,G;IAAjB,C;gFACjC,yB;MAAA,WASqD,wB;MATrD,mB;QASmC,OAAK,oBAAL,SAAK,CAAL,U;MAAA,C;KATnC,C;;MAaqC,W;IAAA,C;oFACrC,yB;MAAA,iC;MgKhLJ,4B;MhKgLI,mB;QASuC,uBgKtL+C,QhKsLnC,SgKtLG,GhKsLW,GgKtLqB,ChKsL/C,C;MAAA,C;KATvC,C;gFAUA,yB;MAAA,6B;MAAA,mB;QASmC,qBAAU,SAAL,GAAiB,GAAtB,C;MAAA,C;KATnC,C;kFAUA,yB;MAAA,WAS6D,wB;MAT7D,+B;MAAA,mB;QASqC,sBAAW,oBAAL,SAAK,CAAL,UAAN,C;MAAA,C;KATrC,C;;MAiBqC,OApDC,SAAL,GAAiB,G;IAoDb,C;;MAOE,OA3DD,SAAL,GAAiB,G;IAA2DX,C;;MAEE,OAAQ,CA7DX,SAAL,GAAiB,GA6DD,Y;IAAR,C;MA/O7C,c;MAG6D,qD;MAH7D,a;IAAA,C;;MAAA,2IAG6D,oCAH7D,G;IAAA,C;wEAmPA,yB;MAAA,+B;MAAA,4B;QAU0C,sBAAM,SAAN,C;MAAA,C;KAV1C,C;0EAWA,yB;MAAA,0B;MAAA,+B;MAAA,4B;QAW2C,sBAAW,OAAL,SAAK,CAAX,C;MAAA,C;KAX3C,C;0EAYA,yB;MAAA,0B;MAAA,+B;MAAA,4B;QAWyC,sBAAW,OAAL,SAAK,CAAX,C;MAAA,C;KAXzC,C;0EAYA,yB;MAAA,0B;MAAA,+B;MAAA,4B;QAW0C,sBAAW,OAAL,SAAK,SAAX,C;MAAA,C;KAX1C,C;;M6BnRqB,sB;IAFrB,C;;MAaiD,07B+OP,U6B/OO,aAAQ,KAAR,C7B+OP,C;I6B/OO,C;;MASzC,aAAQ,KAAR,IAAiB,K7BuIc,K;I6BtInC,C;;MAGsC,OAAA,YAAQ,O;IAAR,C;;MAGmB,+BAAS,YAAT,C;IAAA,C;;MAEF,wB;MAAhC,oB;MACnB,eAAoB,C;IADF,C;;MAEO,sBAAQ,YAAM,O;IAAd,C;;MACgC,Q;MAA9B,IAAI,eAAQ,YAAM,OAAlB,C;QAAA,07B0NO,U6B1NiB,aAAM,mBAAN,EAAM,2BAAN,07B0NjB,C;;Q6B1N+C,MAAM,2BAAuB,YAAM,WAA7B,C;IAA5D,C;;MAMtB,Q;MAAL,IAAI,eAAC,0EAAD,QAAJ,C;QAAiC,OA AO,K;MAExC,OAAe,WAAR,YAAQ,EAAS,07BmHO,K6BnHhB,C;IACnB,C;;MAGY,Q;MAA2B,gBAA3B,gE;MAA2B,c;;QZw4CvB,U;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,aAAO,I;UAAP,e;;QACrB,6B;QAAhB,OA AgB,gBAAhB,C;UAAgB,2B;UYx4C6B,2BZw4CR,OYx4CQ,Q;UAAA,W;YAAuB,oBAAR,YAAQ,EZw4C/B,Oj BzxCF,K6B/GiC,C;;UZw4C9C,IAAI,OAAJ,C;YAAyB,aAAO,K;YAAP,e;;QAC/C,aAAO,I;;MYz4CH,iB;IACJ,C;;MAEkC,OAAA,IAAK,QAAQ,OAAb,KAAqB,C;IAArB,C;;MA7CIC,oD;MACgC,uBAAK,cAAU,IAAV,CAAL,C;MADhC,Y;IAAA,C;MAPJ,OAKqB,qDALrB,M;IAAA,C;;MAAA,c;MAKqB,wD;MALrB,a;IAAA,C;;MAAA,2IAKqB,0CALrB,G;IAAA,C;gFAuDA,yB;MAAA,yC;;QAWsC,wB;UAAW,OAAA,aAAK,KAAL,C7B8FV,K;Q6B9FD,C;MAAA,C;MAXtC,6B;QAWI,OAAO,oBAAW,+BAAU,IAAV,GAAGB,uBAAhB,CAAX,C;MACX,C;KAZA,C;;MAiBqE,e;IAAA,C;;M5BjEjE,4B;MAFwD,gB;IAAnC,C;;MAErB,8B;MAcliBAGmC,SAAK,CAAL,C;MAEnC,iBAGmC,SAAK,EAAL,C;MAEnC,kBAGmC,C;MAEnC,iBAGkC,E;IANtC,C;;MAAA,qC;QAAA,oB;;MAAA,8B;IAAA,C;oGAsBA,yB;MDgLA,6B;MChKA,8C;MAhBA,wB;QAM0D,OAIbQ,YAAY,IAAK,KAAjB,EA A6B,CDkK5D,cCnLsC,KDmL5B,KAAL,GAAiB,GAAtB,CCIK4D,MAA7B,C;MAjBR,C;KAN1D,C;oGAQA,yB;MCyKA,6B;MDjKA,8C;MARA,wB;QAM2D,OASO,YAAY,IAAK,KAAjB,EAA6B,CCmK5D,cD5KuC,KC4K7B,KAAL,GAAiB,KAAtB,CDnK4D,MAA7B,C;MATP,C;KAN3D,C;gGAQA,yB;MAAA,8C;MAAA,wB;QAOKE,m BAAY,IAAK,KAAjB,EAAuB,KAAM,KAA7B,C;MAAA,C;KAPIE,C;oGASA,yB;MAyKA,kBAS6D,sB;MAT7D,+B;M+B1KA,gD;M/BCA,wB;QAM0D,O+BAS,aAAkB,C/B4KhD,eAAW,oBAAL,SAAK,CAAL,iBAAN,C+B5K gD,MAAIB,E/BAGB,K+BAC,KAA9B,C;M/BAT,C;KAN1D,C;0FAQA,yB;MD+IA,6B;MC/IA,wB;QAEsD,OAMD ,cAAK,IAAK,KAAK,GAAW,CDgJ5C,cCtJ6B,KDsJnB,KAAL,GAAiB,GAAtB,CChJ4C,MAAX,IAAf,C;MANC,C;KAFtD,C;0FAGA,yB;MC6IA,6B;MD7IA,wB;QAEuD,OAGF,cAAK,IAAK,KAAK,GAAW,CCiJ5C,cDpJ8B,K CoJpB,KAAL,GAAiB,KAAtB,CDjJ4C,MAAX,IAAf,C;MAHE,C;KAFvD,C;0FAGA,yB;MAAA,6B;MAAA,wB;QAEqD,qBAAK,IAAK,KAAK,GAAK,KAAM,KAAX,IAAf,C;MAAA,C;KAFrD,C;0FAGA,yB;MAwJA,kBAS6D,s

B;MAT7D,+B;MAxJA,wB;QAEuD,O+BAA,eAAW,C/B+J7B,eAAW,oBAAL,SAAK,CAAL,iBAAN,C+B/J6B,M
AAK,K/BAI,K+BAO,KAAK,CAAhB,C;M/BAA,C;KAFvD,C;4FAIA,yB;MDkIA,6B;MCIIA,wB;QAEuD,OAMD,
cAAK,IAAK,KAAK,GAAY,CDmI9C,cCzI+B,KDyIrB,KAAL,GAAiB,GAAtB,CCnI8C,MAAZ,IAAf,C;MANC,C;
KAFvD,C;4FAGA,yB;MCgIA,6B;MDhIA,wB;QAEwD,OAGF,cAAK,IAAK,KAAK,GAAY,CCoI9C,cDvIgC,KCu
ItB,KAAL,GAAiB,KAAtB,CDpI8C,MAAZ,IAAf,C;MAHE,C;KAFxD,C;4FAGA,yB;MAAA,6B;MAAA,wB;QAE
sD,qBAAK,IAAK,KAAK,GAAM,KAAM,KAAZ,IAAf,C;MAAA,C;KAFtD,C;4FAGA,yB;MA2IA,kBAS6D,sB;M
AT7D,+B;MA3IA,wB;QAEwD,O+BAA,eAAW,C/BkI9B,eAAW,oBAAL,SAAK,CAAL,iBAAN,C+BIJ8B,MAAK
,U/BAK,K+BAO,KAAZ,CAAhB,C;M/BAA,C;KAFxD,C;4FAIA,yB;MDqHA,6B;MCRHA,wB;QAEuD,OAMD,cA
Ae,YAAV,IAAK,KAAK,EAAY,CDsH9C,cC5H+B,KD4HrB,KAAL,GAAiB,GAAtB,CCtH8C,MAAZ,CAAf,C;M
ANC,C;KAFvD,C;4FAGA,yB;MCmHA,6B;MDnHA,wB;QAEwD,OAGF,cAAe,YAAV,IAAK,KAAK,EAAY,CCu
H9C,cD1HgC,KCOHtB,KAAL,GAAiB,KAAtB,CDvH8C,MAAZ,CAAf,C;MAHE,C;KAFxD,C;4FAGA,yB;MAA
A,6B;MAAA,wB;QAEsD,qBAaE,YAAV,IAAK,KAAK,EAAM,KAAM,KAAZ,CAAf,C;MAAA,C;KAFtD,C;4FA
GA,yB;MA8HA,kBAS6D,sB;MAT7D,+B;MA9HA,wB;QAEwD,O+BAA,eAAW,C/BqI9B,eAAW,oBAAL,SAAK,
CAAL,iBAAN,C+BrlI8B,MAAK,U/BAK,K+BAO,KAAZ,CAAhB,C;M/BAA,C;KAFxD,C;wFAIA,yB;MDwGA,6
B;MCIGA,4C;MANA,wB;QAEqD,OAMD,WAAW,IAAX,EDyGjB,cC/G2B,KD+GjB,KAAL,GAAiB,GAAtB,CCz
GiB,C;MANC,C;KAFrD,C;wFAGA,yB;MCsGA,6B;MDnGA,4C;MAHA,wB;QAEsD,OAGF,WAAW,IAAX,EC0
GjB,cD7G4B,KC6GIB,KAAL,GAAiB,KAAtB,CD1GiB,C;MAHE,C;KAFtD,C;wFAGA,yB;MAAA,4C;MAAA,w
B;QAEoD,kBAAW,IAAX,EAaiB,KAajB,C;MAAA,C;KAFpD,C;wFAGA,yB;MAiHA,kBAS6D,sB;MAT7D,+B;
M+BjHA,8C;M/BAA,wB;QAEsD,O+BAA,Y/BwHjB,eAAW,oBAAL,SAAK,CAAL,iBAAN,C+BxHiB,E/BAmB,
K+BAnB,C;M/BAA,C;KAFtD,C;wFAIA,yB;MD2FA,6B;MCRFA,kD;MANA,wB;QAEqD,OAMD,cAAc,IAAd,ED
4FjB,cCIG2B,KDkGjB,KAAL,GAAiB,GAAtB,CC5FiB,C;MANC,C;KAFrD,C;wFAGA,yB;MCyFA,6B;MDtFA,k
D;MAHA,wB;QAEsD,OAGF,cAAc,IAAd,EC6FjB,cDhG4B,KCgGIB,KAAL,GAAiB,KAAtB,CD7FiB,C;MAHE,
C;KAFtD,C;wFAGA,yB;MAAA,kD;MAAA,wB;QAEoD,qBAAc,IAAd,EAaOb,KAAPB,C;MAAA,C;KAFpD,C;
wFAGA,yB;MAoGA,kBAS6D,sB;MAT7D,+B;M+BpGA,oD;M/BAA,wB;QAEsD,O+BAA,e/B2GjB,eAAW,oBA
AL,SAAK,CAAL,iBAAN,C+B3GiB,E/BAmB,K+BAnB,C;M/BAA,C;KAFtD,C;wEAlA,yB;MAAA,6B;MAAA,m
B;QAEyC,qBAAK,SAAK,QAaV,C;MAAA,C;KAFzC,C;wEAGA,yB;MAAA,6B;MAAA,mB;QAEyC,qBAAK,S
AAK,QAaV,C;MAAA,C;KAFzC,C;gGAIA,yB;MAAA,8C;MAAA,wB;QAE6D,0BAAU,IAAV,EAAGB,KAhB,
C;MAAA,C;KAF7D,C;wFAIA,yB;MAAA,6B;MAAA,2B;QAEmD,qBAAK,aAAS,QAAd,C;MAAA,C;KAFnD,C;
wFAGA,yB;MAAA,6B;MAAA,2B;QAEmD,qBAAK,cAAU,QAaf,C;MAAA,C;KAFnD,C;wFAGA,yB;MAAA,6
B;MAAA,wB;QAEiD,qBAAK,IAAK,KAAL,GAac,KAAM,KAAzB,C;MAAA,C;KAFjD,C;wFAGA,yB;MAAA,6
B;MAAA,wB;QAEgD,qBAAK,IAAK,KAAL,GAaA,KAAM,KAAxB,C;MAAA,C;KAFhD,C;wFAGA,yB;MAAA,
6B;MAAA,wB;QAEiD,qBAAK,IAAK,KAAL,GAac,KAAM,KAAzB,C;MAAA,C;KAFjD,C;wEAGA,yB;MAAA,
6B;MAAA,mB;QAEgC,qBAAU,CAAL,SAAL,C;MAAA,C;KAFhC,C;8EAIA,yB;MAAA,0B;MAAA,mB;QAUm
C,OAAK,OAAL,SAAK,C;MAAL,C;KAVnC,C;gFAWA,yB;MAAA,4B;MAAA,mB;QAUqC,OAAK,QAAL,SA
AK,C;MAAL,C;KAVrC,C;MAoBiC,gB;IAAA,C;8EACjC,yB;MAAA,kBASqD,sB;MATrD,mB;QASmC,OAAK,oB
AAL,SAAK,CAAL,iB;MAAA,C;KATnC,C;gFAWA,yB;MDoEJ,0B;MAAA,+B;MCPeI,mB;QASqC,ODsEA,eAA
W,OCtEX,SDsEW,CAAX,C;MCEA,C;KATrC,C;kFAUA,yB;MC2DJ,4B;MAAA,iC;MD3DI,mB;QASuC,OC6DA
,gBAAy,QD7DZ,SC6DY,CAAZ,C;MD7DA,C;KATvC,C;MAYmC,W;IAAA,C;gFACnC,yB;MAAA,kBAS6D,sB
;MAT7D,+B;MAAA,mB;QASqC,sBAAW,oBAAL,SAAK,CAAL,iBAAN,C;MAAA,C;KATrC,C;gFAWA,yB;MA
SA,gD;MATA,mB;QAQqC,OAoe,aAAa,SAAb,C;MAPF,C;KARrC,C;kFASA,yB;MAAA,gD;MAAA,mB;QAMu
C,oBAAa,SAAb,C;MAAA,C;KANvC,C;MAQyC,OArDD,oBAAL,SAAK,CAAL,iBAqDe,W;IAAT,C;MAzP7
C,c;MAG4D,qD;MAH5D,a;IAAA,C;MAAA,2IAG4D,oCAH5D,G;IAAA,C;EA6PA,yB;MAAA,6B;MAAA,4B;Q
AWwC,qBAAU,SAAV,C;MAAA,C;KAXxC,C;wEAYA,yB;MAAA,6B;MAAA,4B;QAWyC,qBAAU,SAAV,C;M
AAA,C;KAXzC,C;wEAYA,yB;MAAA,6B;MAAA,4B;QAUuC,qBAAK,SAAL,C;MAAA,C;KAVvC,C;wEAWA,y
B;MAAA,6B;MAAA,4B;QAWwC,qBAAK,SAAK,QAaV,C;MAAA,C;KAXxC,C;uEAaA,yB;MAAA,gD;MAAA,
4B;QASyC,oBAakB,SAaIB,C;MAAA,C;KATzC,C;wEAUA,yB;MAAA,gD;MAAA,4B;QAS0C,oBAAa,SAAb,C
;MAAA,C;KAT1C,C;M6BpTqB,sB;IAFrB,C;MAagD,O7BiRT,S6BjRS,aAAQ,KAAR,C7BiRT,C;I6BjRS,C;MA
SxC,aAAQ,KAAR,IAaiB,K7BmKY,K;I6BIKjC,C;MAGsC,OAAA,YAAQ,O;IAAR,C;MAGkB,8BAAS,YAAT,

C;IAAA,C;;MAEF,uB;MAA/B,oB;MACnB,eAAoB,C;IADF,C;;MAEO,sBAAQ,YAAM,O;IAAd,C;;MAC+B,Q;M
AA9B,IAAI,eAAQ,YAAM,OAAIB,C;QAAA,O7B4PK,S6B5PmB,aAAM,mBAAN,EAAM,2BAAN,O7B4PnB,C;;
Q6B5PgD,MAAM,2BAAuB,YAAM,WAA7B,C;IAA3D,C;;MAMrB,Q;MAAL,IAAI,eAAC,0EAAD,OAAJ,C;QA
AgC,OAAO,K;MAEvC,OAAe,WAAR,YAAQ,EAAS,O7B+IK,K6B/Id,C;IACnB,C;;MAGY,Q;MAA2B,gBAA3B,
gE;MAA2B,c;;Qbw4CvB,U;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,aAAO,I;UAAP,e;;QACrB,6B;QAAhB,
OAAgB,gBAAhB,C;UAAgB,2B;Uax4C6B,2Bbw4CR,Oax4CQ,O;UAAA,W;YAAsB,oBAAR,YAAQ,Ebw4C9B,O
hb7vCJ,K6B3IkC,C;;Ubw4C7C,IAAI,OAAJ,C;YAAyB,aAAO,K;YAAP,e;;QAC/C,aAAO,I;;Maz4CH,iB;IACJ,C
;;MAEKc,OAAA,IAAK,QAAQ,OAAb,KAAqB,C;IAArB,C;;MA7CIC,mD;MACgC,sBAAK,eAAS,IAAT,CAAL,
C;MADhC,Y;IAAA,C;;;MAPJ,OAKqB,oDALrB,M;IAAA,C;;MAAA,c;MAKqB,wD;MALrB,a;IAAA,C;;MAAA
,2IAKqB,0CALrB,G;IAAA,C;8EAuDA,yB;MAAA,uC;;QAWoC,wB;UAAW,OAAA,aAAK,KAAL,C7B0HV,K;Q
6B1HD,C;MAAA,C;MAXpC,6B;QAWI,OAAO,mBAAU,gCAAS,IAAT,GAAe,sBAAf,CAAV,C;MACX,C;KAZA
,C;;MAiBkE,e;IAAA,C;;M8J9C9D,iC;MAjBsD,2BAAgB,KAAhB,EAAuB,YAAvB,EAAqC,CAArC,C;IAApC,C;;
MACe,iB;IAAA,C;;MACO,gB;IAAA,C;;MAEM,W3LuCoB,Y2LvCpB,U3LuCqC,KAAjB,E2LvCX,K3LuCwC,K
AA7B,C2LvCpB,K;MAAA,S;QAAkB,O3LuCE,Y2LvCF,K3LuCmB,KAAjB,E2LvCO,S3LuCsB,KAA7B,C2LvCF
,K;;MAAIB,W;IAAA,C;;MAEZ,O3LqCgC,Y2LrChC,U3LqCiD,KAAjB,E2LrCxB,S3LqCqD,KAA7B,C2LrChC,I;I
AAA,C;;MAItB,UAAwB,M;MADhC,2CAAuB,kBAaA,KAAM,UAAAnB,KACf,2CAAS,KAAM,MAAf,cAAwB,6C
AAQ,KAAM,KAAAd,QAAxB,CADe,CAAvB,C;IAAA,C;;MAIA,OAAI,cAAJ,GAAe,EAAf,GAAwB,MAAK,U3Ls
KA,K2LtKL,QAAqB,S3LsKhB,K2LtKL,I;IAAxB,C;;MAE8B,OAAE,UAAF,qBAAU,S;IAAV,C;;MAEIC,mC;MA
CI,aAC8B,cAAU,4BAAK,UAAf,EAA0B,4BAAK,UAA/B,C;IAFIC,C;;;MAAA,0C;QAAA,yB;;MAAA,mC;IAAA
,C;;MAmDA,uC;MAjCI,IAAI,SAAQ,CAAZ,C;QAAuB,MAAa,gCAAyB,wBAAzB,C;MACpC,IAAI,SAAQ,WA
AZ,C;QAA2B,MAAa,gCAAyB,wEAAzB,C;MAG5C,aAGyB,K;MAEzB,YAGwB,4BAA0B,KAA1B,EAAiC,YAA
jC,EAA+C,IAA/C,C;MAExB,YAGuB,I;IAvB3B,C;;MAyB4C,mCAAwB,UAAxB,EAA+B,SAA/B,EAAqC,SAAr
C,C;IAAA,C;;MAGH,OAAI,YAAO,CAAX,G3Ld6B,Y2Lcf,U3LdgC,KAAjB,E2LcP,S3LdoC,KAA7B,C2Lcf,IAA
d,G3Ld6B,Y2LcG,U3Ldc,KAAjB,E2LcW,S3LdkB,KAA7B,C2LcG,I;IAAhC,C;;MAIzB,UAAwB,M;MADhC,iDA
A6B,kBAaA,KAAM,UAAAnB,KACrB,2CAAS,KAAM,MAAf,cAAwB,6CAAQ,KAAM,KAAAd,QAAxB,KAA8C,c
AAQ,KAAM,KADvC,CAA7B,C;IAAA,C;;MAIA,OAAI,cAAJ,GAAe,EAAf,GAAwB,OAAM,MAAK,U3LmHN,
K2LnHC,QAAqB,S3LmHtB,K2LnHC,IAAN,SAAGD,SAAhD,I;IAAxB,C;;MAE8B,OAAI,YAAO,CAAX,GAAgB
,UAAF,qBAAU,SAAV,cAAqB,SAAnC,GAAgD,UAAF,2BAAgB,SAAhB,eAA4B,CAAC,SAAD,IAA5B,C;IAA9
C,C;;MAEIC,yC;IAAA,C;;MAS+F,2BAAgB,UAAhB,EAA4B,QAA5B,EAA5C,IAAtC,C;IAAA,C;;;MAT/F,gD;Q
AAA,+B;;MAAA,yC;IAAA,C;;MAoBwE,uB;MACxE,sBAA2B,I;MAC3B,iBAAmC,OAAO,CAA1C,G3L/CkE,Y
2L+CrB,K3L/CsC,KAAjB,E2L+CZ,I3L/CyC,KAA7B,C2L+CrB,KAA7C,G3L/CkE,Y2L+CF,K3L/CmB,KAAjB,E
2L+CO,I3L/CsB,KAA7B,C2L+CF,K;MACHe,c3L6LmC,S2L7LhB,I3L6LgB,C;M2L5LnC,cAAuB,cAAJ,GAAa,K
AAb,GAAwB,mB;IAJV,C;;MAMC,qB;IAAA,C;;MAG9B,YAAy,W;MACZ,IAAI,6BAAS,mBAAT,QAAJ,C;QA
CI,IAAI,CAAC,cAAL,C;UAAc,MAAa,6B;QAC3B,iBAAU,K;;QAEV,c3LzC6C,S2LyC7C,W3LzCuD,KAAK,G2
LyCpD,W3LzC+D,KAAX,IAAf,C;;M2L2CjD,OAAO,K;IACX,C;;;ICIHJ,C;;MAIgc,uB;IAAA,C;;IAMhC,C;;MA
Igc,wB;IAAA,C;;IAMhC,C;;MAIgc,sB;IAAA,C;;IAMhC,C;;MAIgc,uB;IAAA,C;;M7J3B5B,6B;MAFyD,gB;IA
AnC,C;;MAEtB,+B;MACI,iBAGoC,a;MAEpC,iBAGoC,c;MAEpC,kBAGmC,C;MAEnC,iBAGkC,E;IANBtC,C;;;
MAAA,sC;QAAA,qB;;MAAA,+B;IAAA,C;sGAsBA,yB;MhC0LA,WAS6D,wB;MAT7D,+B;MgCIKA,gD;MAxB
A,wB;QAM0D,OAYBS,aAAa,IAAK,KAAIB,EAA8B,ChCoK5D,eAAW,oBgC7LyB,KhC6L9B,KAAK,CAAL,UA
AN,CgCpK4D,MAA9B,C;MAzBT,C;KAN1D,C;sGAQA,yB;M9BmLA,aAS6D,0B;MAT7D,+B;M8BnKA,gD;MA
hBA,wB;QAM2D,OAIbQ,aAAa,IAAK,KAAIB,EAA8B,C9BqK5D,eAAW,oB8BtL0B,K9BsL/B,KAAK,CAAL,Y
AAN,C8BrK4D,MAA9B,C;MAjBR,C;KAN3D,C;sGAQA,yB;M/BkLA,kBAS6D,sB;MAT7D,+B;M+B1KA,gD;M
ARA,wB;QAMyD,OASU,aAAa,IAAK,KAAIB,EAA8B,C/B4K5D,eAAW,oB+BrLwB,K/BqL7B,KAAK,CAAL,iB
AAN,C+B5K4D,MAA9B,C;MATV,C;KANzD,C;kGAQA,yB;MAAA,gD;MAAA,wB;QAOmE,oBAAa,IAAK,KA
AIB,EAAwB,KAAM,KAA9B,C;MAAA,C;KAPnE,C;4FASA,yB;MhCyJA,WAS6D,wB;MAT7D,+B;MgCzJA,wB;
QAEuD,OASA,eAAM,IAAK,KAAK,KAAW,ChCuJ7C,eAAW,oBgChKiB,KhCgKtB,KAAK,CAAL,UAAN,CgCv
J6C,MAAX,CAAhB,C;MATA,C;KAFvD,C;4FAGA,yB;M9BuJA,aAS6D,0B;MAT7D,+B;M8BvJA,wB;QAEwD,
OAMD,eAAM,IAAK,KAAK,KAAW,C9BwJ7C,eAAW,oB8B9JkB,K9B8JvB,KAAK,CAAL,YAAN,C8BxJ6C,M

AAX,CAAhB,C;MANC,C;KAFxD,C;4FAGA,yB;M/B2JA,kBAS6D,sB;MAT7D,+B;M+B3JA,wB;QAEsD,OAGC,
eAAM,IAAK,KAAK,KA AW,C/B+J7C,eAAW,oB+BIKgB,K/BkKrB,KAAK,CAAL,iBAAN,C+B/J6C,MAAX,CA
AhB,C;MAHD,C;KAFtD,C;4FAGA,yB;MAAA,+B;MAAA,wB;QAEuD,sBAAM,IAAK,KAAK,KAAK,KAAM,K
AAX,CAAhB,C;MAAA,C;KAFvD,C;8FAIA,yB;MhC4IA,WAS6D,wB;MAT7D,+B;MgC5IA,wB;QAEwD,OASA,
eAAM,IAAK,KAAK,UAA Y,ChC0I/C,eAAW,oBgCnJmB,KhCmJxB,KAAK,CAAL,UAAN,CgC1I+C,MAAZ,CA
AhB,C;MATA,C;KAFxD,C;8FAGA,yB;M9B0IA,aAS6D,0B;MAT7D,+B;M8B1IA,wB;QAEyD,OAMD,eAAM,IA
AK,KAAK,UAA Y,C9B2I/C,eAAW,oB8BjJoB,K9BiJzB,KAAK,CAAL,YAAN,C8B3I+C,MAAZ,CAAhB,C;MAN
C,C;KAFzD,C;8FAGA,yB;M/B8IA,kBAS6D,sB;MAT7D,+B;M+B9IA,wB;QAEuD,OAGC,eAAM,IAAK,KAAK,
UAA Y,C/BkJ/C,eAAW,oB+BrJkB,K/BqJvB,KAAK,CAAL,iBAAN,C+BIJ+C,MAAZ,CAAhB,C;MAHD,C;KAFv
D,C;8FAGA,yB;MAAA,+B;MAAA,wB;QAEwD,sBAAM,IAAK,KAAK,UAA M,KAAZ,CAAhB,C;MAA
A,C;KAFxD,C;8FAIA,yB;MhC+HA,WAS6D,wB;MAT7D,+B;MgC/HA,wB;QAEwD,OASA,eAAM,IAAK,KAA
K,UAA Y,ChC6H/C,eAAW,oBgCtImB,KhCsIxB,KAAK,CAAL,UAAN,CgC7H+C,MAAZ,CAAhB,C;MATA,C;K
AFxD,C;8FAGA,yB;M9B6HA,aAS6D,0B;MAT7D,+B;M8B7HA,wB;QAEyD,OAMD,eAAM,IAAK,KAAK,UAA
Y,C9B8H/C,eAAW,oB8BpIoB,K9BoIzB,KAAK,CAAL,YAAN,C8B9H+C,MAAZ,CAAhB,C;MANC,C;KAFzD,C
;8FAGA,yB;M/BiIA,kBAS6D,sB;MAT7D,+B;M+BjIA,wB;QAEuD,OAGC,eAAM,IAAK,KAAK,UAA Y,C/BqI/C,
eAAW,oB+BxIkB,K/BwIvB,KAAK,CAAL,iBAAN,C+BrI+C,MAAZ,CAAhB,C;MAHD,C;KAFvD,C;8FAGA,yB;
MAAA,+B;MAAA,wB;QAEwD,sBAAM,IAAK,KAAK,UAA M,KAAZ,CAAhB,C;MAAA,C;KAFxD,C;0
FAIA,yB;MhCkHA,WAS6D,wB;MAT7D,+B;MgCzGA,8C;MATA,wB;QAEsD,OASA,YAA Y,IAAZ,EhCgHjB,e
AAW,oBgCzHe,KhCyHpB,KAAK,CAAL,UAAN,CgChHiB,C;MATA,C;KAFtD,C;0FAGA,yB;M9BghA,aAS6D,
0B;MAT7D,+B;M8B1GA,8C;MANA,wB;QAEuD,OAMD,YAA Y,IAAZ,E9BiHjB,eAAW,oB8BvHgB,K9BuHrB,
KAAK,CAAL,YAAN,C8BjHiB,C;MANC,C;KAFvD,C;0FAGA,yB;M/BoHA,kBAS6D,sB;MAT7D,+B;M+BjHA,
8C;MAHA,wB;QAEqD,OAGC,YAA Y,IAAZ,E/BwHjB,eAAW,oB+B3Hc,K/B2HnB,KAAK,CAAL,iBAAN,C+Bx
HiB,C;MAHD,C;KAFrD,C;0FAGA,yB;MAAA,8C;MAAA,wB;QAEsD,mBAAY,IAAZ,EAakB,KAAIB,C;MAAA
,C;KAFtD,C;0FAIA,yB;MhCqGA,WAS6D,wB;MAT7D,+B;MgC5FA,oD;MATA,wB;QAEsD,OASA,eAAe,IAAf,
EhCmGjB,eAAW,oBgC5Ge,KhC4GpB,KAAK,CAAL,UAAN,CgCnGiB,C;MATA,C;KAFtD,C;0FAGA,yB;M9B
mGA,aAS6D,0B;MAT7D,+B;M8B7FA,oD;MANA,wB;QAEuD,OAMD,eAAe,IAAf,E9BoGjB,eAAW,oB8B1GgB
,K9B0GrB,KAAK,CAAL,YAAN,C8BpGiB,C;MANC,C;KAFvD,C;0FAGA,yB;M/BuGA,kBAS6D,sB;MAT7D,+B
;M+BpGA,oD;MAHA,wB;QAEqD,OAGC,eAAe,IAAf,E/B2GjB,eAAW,oB+B9Gc,K/B8GnB,KAAK,CAAL,iBAA
N,C+B3GiB,C;MAHD,C;KAFrD,C;0FAGA,yB;MAAA,oD;MAAA,wB;QAEsD,sBA Ae,IAAf,EAaqB,KAArB,C;
MAAA,C;KAFtD,C;0EAIA,yB;MAAA,+B;MAAA,mB;QAE0C,sBAAM,SAAK,MAAX,C;MAAA,C;KAF1C,C;0
EAGA,yB;MAAA,+B;MAAA,mB;QAE0C,sBAAM,SAAK,MAAX,C;MAAA,C;KAF1C,C;kGAIA,yB;MAAA,gD;
MAAA,wB;QAE+D,2BAAW,IAAX,EAAiB,KAAjB,C;MAAA,C;KAF/D,C;0FAIA,yB;MAAA,+B;MAAA,2B;QA
EoD,sBAAM,oBAAS,QAAT,CAAN,C;MAAA,C;KAFpD,C;0FAGA,yB;MAAA,+B;MAAA,2B;QAEoD,sBAAM,
6BAAU,QA AV,CAAN,C;MAAA,C;KAFpD,C;0FAGA,yB;MAAA,+B;MAAA,wB;QAE mD,sBAAM,IAAK,KAA
L,KAAc,KAAM,KAAPB,CAAN,C;MAAA,C;KAFnD,C;wFAGA,yB;MAAA,+B;MAAA,wB;QAEkD,sBAAM,IA
AK,KAAL,IAAa,KAAM,KAAnB,CAAN,C;MAAA,C;KAFID,C;0FAGA,yB;MAAA,+B;MAAA,wB;QAE mD,sB
AAM,IAAK,KAAL,KAAc,KAAM,KAAPB,CAAN,C;MAAA,C;KAFnD,C;0EAGA,yB;MAAA,+B;MAAA,mB;Q
AEiC,sBAAM,SAAK,MAAX,C;MAAA,C;KAFjC,C;gFAIA,yB;MAAA,0B;MAAA,mB;QAUmC,OAAK,OAL,S
AAK,S;MAAL,C;KAVnC,C;kFAWA,yB;MAAA,4B;MAAA,mB;QAUqC,OAAK,QAAL,SAAK,S;MAAL,C;KAV
rC,C;MAqBiC,OAAA,SAAK,Q;IAAL,C;MAUE,gB;IAAA,C;kFAEnC,yB;MhC+EJ,0B;MAAA,+B;MgC/El,mB;
QASqC,OhCiFC,eAAW,OgCjFZ,ShCiFY,SAAX,C;MgCjFD,C;KATrC,C;oFAUA,yB;M9BsEJ,4B;MAAA,iC;M8
BtEI,mB;QASuC,O9BwEC,gBAAY,Q8BxEb,S9BwEa,SAAZ,C;M8BxED,C;KATvC,C;gFAUA,yB;M/BqEJ,6B;M
+BrEI,mB;QASmC,O/BuEC,c+BvED,S/BuEW,QA AV,C;M+BvED,C;KATnC,C;MAYqC,W;IAAA,C;kFAErC,y
B;MASA,kD;MATA,mB;QAQqC,OASE,cAAc,SAAd,C;MATF,C;KARrC,C;oFASA,yB;MAAA,kD;MAAA,mB;
QAQuC,qBAAc,SAAd,C;MAAA,C;KARvC,C;MAUyC,qBAAc,SAAd,C;IAAA,C;MA5P7C,c;MAG6D,qD;M
AH7D,a;IAAA,C;MAAA,2IAG6D,oCAH7D,G;IAAA,C;wEAgQA,yB;MAAA,+B;MAAA,4B;QAW0C,sBAAW,o
BAAL,SAAK,CAAX,C;MAAA,C;KAX1C,C;0EAYA,yB;MAAA,+B;MAAA,4B;QAW2C,sBAAW,oBAAL,SAA
K,CAAX,C;MAAA,C;KAX3C,C;0EAYA,yB;MAAA,+B;MAAA,4B;QAWyC,sBAAW,oBAAL,SAAK,CAAX,C;

MAAA,C;KAXzC,C;0EAYA,yB;MAAA,+B;MAAA,4B;QAU0C,sBAAM,SAAN,C;MAAA,C;KAV1C,C;yEAYA,yB;MAAA,kD;MAAA,4B;QAS2C,qBAAMb,SAANb,C;MAAA,C;KAT3C,C;0EAUa,yB;MAAA,kD;MAAA,4B;QAS4C,qBAAc,SAAd,C;MAAA,C;KAT5C,C;;MDvTqB,sB;IAFrB,C;;MAaiD,OCgSP,UDhSO,aAAQ,KAAR,CCgSP,C;IDhSO,C;;MASzC,aAAQ,KAAR,IAAiB,KC8Kc,K;ID7KnC,C;;MAGsC,OAAA,YAAQ,O;IAAR,C;;MAGmB,+BAAS,YAAT,C;IAAA,C;;MAEF,wB;MAAhC,oB;MACnB,eAAoB,C;IADF,C;;MAEO,sBAAQ,YAAM,O;IAAd,C;;MACgC,Q;MAA9B,IAAI,eAAQ,YAAM,OAAIB,C;QAAA,OC2QO,UD3QiB,aAAM,mBAAN,EAAM,2BAAN,OC2QjB,C;;;QD3Q+C,MAAM,2BAAuB,YAAM,WAA7B,C;IAA5D,C;;;MAMtB,Q;MAAL,IAAI,eAAC,0EAAD,QAAJ,C;QAAiC,OAAO,K;MAExC,OAAe,WAAR,YAAQ,EAAS,OC0JO,KD1JhB,C;IACnB,C;;MAGY,Q;MAA2B,gBAA3B,gE;MAA2B,c;;Qdw4CvB,U;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,aAAO,I;UAAp,e;;QACrB,6B;QAAhB,OAAgB,gBAAhB,C;UAAgB,2B;Ucx4C6B,2Bdw4CR,Ocx4CQ,Q;UAAA,W;YAAuB,oBAAR,YAAQ,Edw4C/B,OelvCF,KDtJiC,C;;Udw4C9C,IAAI,OAAJ,C;YAAyB,aAAO,K;YAAP,e;;;QAC/C,aAAO,I;;;Mcz4CH,iB;IACJ,C;;MAEkC,OAAA,IAAK,QAAQ,OAAb,KAAqB,C;IAArB,C;;;MA7CIC,oD;MACgC,uBAAK,iBAAU,IAAV,CAAL,C;MADhC,Y;IAAA,C;;;MAPJ,OAKqB,qDALrB,M;IAAA,C;;MAAA,c;MAKqB,wD;MALrB,a;IAAA,C;;MAAA,2IAKqB,0CALrB,G;IAAA,C;gFAuDA,yB;MAAA,yC;;QAWsC,wB;UAAW,OAAA,aAAK,KAAL,CCqIV,K;QDrID,C;MAAA,C;MAXtC,6B;QAWI,OAAO,oBAAW,kBAAU,IAAV,EAAGb,uBAAhB,CAAX,C;MACX,C;KAZA,C;;MAiBqE,e;IAAA,C;;M+J9CjE,kC;MAJyBd,4BAAiB,KAAjB,EAawB,YAAxB,K;IAAtC,C;;MAce,iB;IAAA,C;;MACO,gB;IAAA,C;;MAEM,W9J+CoB,a8J/CpB,U9J+CSc,KAAIB,E8J/CX,K9J+CyC,KAA9B,C8J/CpB,K;MAAA,S;QAAkB,O9J+CE,a8J/CF,K9J+CoB,KAAIB,E8J/CO,S9J+CuB,KAA9B,C8J/CF,K;;MAAIB,W;IAAA,C;;MAEb,O9J6CiC,a8J7CjC,U9J6CmD,KAAIB,E8J7CzB,S9J6CuD,KAA9B,C8J7CjC,I;IAAA,C;;MAItB,UAAwB,M;MADhC,8CAAwB,kBAAa,KAAM,UAAAnB,KACb,2CAAS,KAAM,MAAf,cAAwB,6CAAQ,KAAM,KAAd,QAAxB,CADgB,CAAxB,C;IAAA,C;;MAIA,OAAI,cAAJ,GAAe,EAaf,GAAwB,M9JuKK,CArCb,U8JIIjB,U9JkI4B,KAAL,KAAoB,CATzB,U8JzHP,U9JyHa,yB8JzHH,E9JyHG,CAAN,CASyB,MAApB,CAAN,CAqCIB,MAAK,Q8JvKV,Q9JuKK,CArCb,U8JIIoB,S9JkIT,KAAL,KAAoB,CATzB,U8JzH6B,S9JyHvB,yB8JzHgC,E9JyHhC,CAAN,CASyB,MAApB,CAAN,CAqCIB,MAAK,Q8JvKV,I;IAAxB,C;;MAE8B,OAAE,UAAF,qBAAU,S;IAAV,C;;MAEIC,oC;MACI,aAC+B,iBAAW,6BAAM,UAAjB,EAA4B,6BAAM,UAAIC,C;IAFnC,C;;;MAAA,2C;QAAA,0B;;MAAA,oC;IAAA,C;;;MamDA,wC;MAjCI,IAAI,gBAAJ,C;QAAwB,MAAa,gCAAyB,wBAAzB,C;MACrC,IAAI,sCAAJ,C;QAA4B,MAAa,gCAAyB,yEAAzB,C;MAG7C,aAG0B,K;MAE1B,YAGyB,4BAA0B,KAA1B,EA AiC,YAAjC,EAA+C,IAA/C,C;MAEzB,YAGwB,I;IAvB5B,C;;MAyB6C,oCAAyB,UAAzB,EAAgC,SAAhC,EAAsC,SAAtC,C;IAAA,C;;MAGJ,OAAI,uBAAO,CAAX,G9JN8B,a8JMhB,U9JNkC,KAAIB,E8JMR,S9JNsC,KAA9B,C8JMhB,IAAd,G9JN8B,a8JME,U9JNgB,KAAIB,E8JMU,S9JNoB,KAA9B,C8JME,I;IAAhC,C;;MAIzB,UAAwB,M;MADhC,kDAA8B,kBAAa,KAAM,UAAAnB,KACtB,2CAAS,KAAM,MAAf,cAAwB,6CAAQ,KAAM,KAAd,QAAxB,KAA8C,kBAAQ,KAAM,KAAd,CADxB,CAA9B,C;IAAA,C;;MAIA,OAAI,cAAJ,GAAe,EAaf,GAAwB,OAAAM,M9JoHD,CArCb,U8J/EX,U9J+EsB,KAAL,KAAoB,CATzB,U8JtED,U9JsEO,yB8JtEG,E9JsEH,CAAN,CASyB,MAApB,CAAN,CAqCIB,MAAK,Q8JpHJ,Q9JoHD,CArCb,U8J/E0B,S9J+Ef,KAAL,KAAoB,CATzB,U8JtEmC,S9JsE7B,yB8JtEsC,E9JsEtC,CAAN,CASyB,MAApB,CAAN,CAqCIB,MAAK,Q8JpHJ,IAAN,SAAqf,cAAU,6BAAU,EAAV,CAAV,CAAyB,QAA9G,I;IAAxB,C;;MAE8B,OAAI,uBAAO,CAAX,GAAgB,UAAF,qBAAU,SAAV,cAAqB,SAArB,WAAAd,GAAgD,UAAF,2BAAgB,SAAhB,cAA6B,SAAD,aAA5B,W;IAA9C,C;;MAEIC,0C;IAAA,C;;MASmG,4BAAiB,UAAjB,EAA6B,QAA7B,EAAuC,IAAvC,C;IAAA,C;;;MATnG,iD;QAAA,gC;;MAAA,0C;IAAA,C;;;MAoB4E,wB;MAC5E,sBAA2B,I;MAC3B,iBAAmC,kBAAO,CAA1C,G9JvCmE,a8JuCtB,K9JvCwC,KAAIB,E8JuCb,I9JvC2C,KAA9B,C8JuCtB,KAA7C,G9JvCmE,a8JuCH,K9JvCqB,KAAIB,E8JuCM,I9JvCwB,KAA9B,C8JuCH,K;MACHe,c9J4MsC,U8J5MnB,I9J4MmB,C;M8J3MtC,cAAuB,cAAJ,GAAa,KAAb,GAAwB,mB;IAJT,C;;MAMA,qB;IAAA,C;;MAG9B,YAAy,W;MACZ,IAAI,6BAAS,mBAAT,QAAJ,C;QACI,IAAI,CAAC,cAALL,C;UAAc,MAAa,6B;QAC3B,iBAAU,K;;;QAEV,c9JtC+C,U8JsC/C,W9JtC0D,KAAK,K8JsCvD,W9JtCkE,KAAX,CAAhB,C;;M8JwCnD,OAAO,K;IACX,C;;wECpHJ,yB;MAAA,8C;MAAA,uB;QAOI,OAAO,MAAM,CAAN,EAAS,CAAT,C;MACX,C;KARA,C;wEAUa,yB;MAAA,8C;MAAA,uB;QAOI,OAAO,MAAM,CAAN,EAAS,CAAT,C;MACX,C;KARA,C;wEAUa,yB;MAAA,8C;MAAA,uB;QAOI,OAAO,MAAM,CAAN,EAAS,CAAT,C;MACX,C;KARA,C;;MvL5BkD,OAAuB,0BAAtB,KAAO,WAAe,EAAU,KAAO,WAAjB,C;IAAvB,C;;MAEG,OAAc,gCAAuB,iBA

AU,gCAAV,C;IAAxB,C;;MAGD,OPwSZ,SAvGI,oBOjMS,EPiMd,KAAK,CAAL,iBOjMc,KPiMT,oBOjMuB,EPi
M5B,KAAK,CAAL,iBOjMc,CPwSH,QAAV,C;IOxSY,C;;MAEG,OPsSf,SAvGI,oBO/LY,EP+LjB,KAAK,CAAL,i
BO/LiB,QP+LZ,oBO/L0B,EP+L/B,KAAK,CAAL,iBO/LiB,CPsSN,QAAV,C;IOtSe,C;;MAOnD,eAAe,EwByLoB,
K;MxBxLnC,cAAc,EwBwLqB,K;MxBvLnC,IAAI,qBAAU,CAAd,C;QACI,OwB2C+D,axB3CpD,EwB2CsE,KAA
IB,ExB3C/C,EwB2C6E,KAA9B,CxB3CpD,IAAJ,GAAa,aAAb,GAA2B,a;;MAItC,IAAI,uBAAy,CAAhB,C;QACI,
OAAO,UAAm,aAAW,OAAx,CAAN,C;;MAIX,eAAiB,4BAAc,CAAd,CAAD,KAAoB,OAApB,CAAD,WAAkC,
CAAIC,C;MACf,UAAU,kBAAW,kBAAW,OAAx,CAAX,C;MACV,OAAO,UAAm,iCwBgCsD,aAAkB,CxBhCz
D,UAAm,GAAN,CwBgCyD,MAAIB,EAA8B,CxBhCvD,UAAm,OAAN,CwBgCuD,MAA9B,CxBhCvC,KAAJ,G
AAkC,CAAIC,GAAyC,CAApD,EAAN,C;IAEX,C;;MAOe,Q;MAHX,eAAe,EwBqKoB,K;MxBpKnC,cAAc,EwBo
KqB,K;MxBnKnC,IAAI,qBAAU,CAAd,C;QACW,IwBuBwD,axBvBpD,EwBuBsE,KAAIB,ExBvB/C,EwBuB6E,
KAA9B,CxBvBpD,IAAJ,C;UACH,S;;;UAEA,OwB8CgD,UxB9ChD,EwB8C2D,KAAK,UxB9C3D,EwB8CuE,KA
AZ,CAAhB,C;;QxBjDpD,W;;MAQJ,IAAI,uBAAy,CAAhB,C;QACI,OAAO,UAAm,gBAAW,OAAx,CAAN,C;;
MAIX,eAAiB,4BAAc,CAAd,CAAD,KAAoB,OAApB,CAAD,WAAkC,CAAIC,C;MACf,UAAU,kBAAW,kBAA
W,OAAx,CAAX,C;MACV,OAAO,UAAm,awBQsD,aAAkB,CxBR9D,UAAm,GAAN,CwBQ8D,MAAIB,EAA8B
,CxBR5D,UAAm,OAAN,CwBQ4D,MAA9B,CxBR5C,KAAJ,GAAkC,OAAIC,KAAN,CAAN,C;IACX,C;;MAIL,I
AAE,QAAF,CAAE,CAAF,C;QADyC,OAC5B,W;;QACb,SP+LuC,aO/LIC,4BAAK,UP+L0C,KAAb,CO/LvC,C;U
AFyC,OAEP,4BAAK,U;;UACvC,SP8LuC,aO9LIC,4BAAK,UP8L0C,KAAb,CO9LvC,C;YAHyC,OAGP,4BAAK,
U;eACvC,SAAK,UAAAL,C;YAJyC,OPyON,SOxOX,YAAF,CAAE,CPqOW,C;;;YOzOM,OPcY,SAAU,CA2N5B,S
OpOP,YAAAnB,IAAI,UAAe,CPoOO,CA3N4B,MAAK,GAAW,CA2N5C,SOpOY,UPoOZ,CA3N4C,MAAX,IAAf,
C;;;IOdZ,C;;MAUzC,IAAE,QAAF,CAAE,CAAF,C;QAD2C,OAC9B,2B;;QACb,SwByLuC,cxBzLIC,6BAAM,Uw
ByL0C,KAAAd,CxBzLvC,C;UAF2C,OAER,6BAAM,U;;UACzC,SwBwLuC,cxBzLIC,6BAAM,UwBwL0C,KAAAd,
CxBzLvC,C;YAH2C,OAGR,6BAAM,U;eACzC,4C;YAJ2C,OwB+OL,UxB3Od,uBAAF,CAAE,CwB2Oc,C;;YxB/
OK,OwBQY,UAAW,CAuO5B,UxBxOF,uBAA3B,IAAI,oBAAuB,CwBwOE,CAvO4B,MAAK,KAAW,CxBdzB,g
CwBCyB,MAAX,CAAhB,C;;;IxBRZ,C;;MAYH,QAAC,CAAqB,GAaf,UAAp,IAAmC,CAAC,MAAO,EAAW,IA
AJ,EAAf,IAAgC,C;IAAnE,C;;MAGE,OAAC,qBAAO,EAAP,CAAW,WAAZ,GAAyB,IAAzB,GAAiC,YAAjC,W;I
AAA,C;;MAGA,uBAAc,CAAd,EAAiB,EAAjB,C;IAAA,C;;MAG1C,IAAI,gBAAK,CAAT,C;QAAY,OAAS,WAA
F,CAAE,EAAS,IAAT,C;MAErB,eAAiB,qBAAO,CAAP,CAAD,yBAAa,IAAb,EAAD,WAAwB,CAAxB,C;MACf,
UAAU,WAAI,sCAAW,IAAX,EAAJ,C;MACV,IAAI,kBAAO,IAAX,C;QACI,uCAAo,IAAP,E;QACA,4CAAY,C
AAZ,E;;MAEJ,OAAgB,WAAAT,QAAS,EAAS,IAAT,CAAT,GAA8B,WAAJ,GAAL,EAAS,IAAT,C;IACzC,C;oFw
LhGA,yB;MAAA,gD;MAAA,4B;QAO6C,OAAQ,a/LsLhB,c+LlLgB,C;MAAR,C;KAP7C,C;oGASA,yB;MtFuCA,i
B;MsFvCA,4B;QAOqD,OtFuCqB,azGsIrC,cyGtlqC,C;MsFvCrB,C;KAPrD,C;sGASA,yB;MAAA,kE;MAAA,4B;Q
AOsD,OAAQ,sB/LoKzB,c+LpKyB,C;MAAR,C;KAPtD,C;8FASA,yB;MAAA,0D;M/L6PA,6B;M+L7PA,4B;QAQ
mD,O/L+PZ,c+L/PoB,kB/L0JtB,c+LlJsB,C/L+PpB,C;M+L/PY,C;KARnD,C;4FAUA,yB;MAAA,wD;M/LmPA,6B
;M+LnPA,4B;QAQkD,O/LqPX,c+LrPmB,iB/LgJrB,c+LhJqB,C/LqPnB,C;M+LrPW,C;KARID,C;gFAUA,yB;MA
AA,4C;M/LyOA,6B;M+LzOA,sC;QAcYD,O/LqOIB,c+LrO0B,W/LgI5B,c+LhI4B,EAAW,QAAX,C/LqO1B,C;M+
LrOkB,C;KAdzD,C;kFAiBA,yB;MAAA,8C;M/LwNA,6B;M+LxNA,sC;QAc0D,O/LoNnB,c+LpN2B,Y/L+G7B,c+
L/G6B,EAAy,QAAX,C/LoN3B,C;M+LpNmB,C;KAd1D,C;oFAiBA,yB;MAAA,gD;MAAA,4B;QAO8C,OAAS,ah
KgHhB,cgKhHgB,C;MAAT,C;KAP9C,C;oGASA,yB;MAAA,gE;MAAA,4B;QAOsD,OAAS,qBhKuGxB,cgKvGw
B,C;MAAT,C;KAPtD,C;sGASA,yB;MAAA,kE;MAAA,4B;QAOuD,OAAS,sBhK8FzB,cgK9FyB,C;MAAT,C;KA
PvD,C;8FASA,yB;MAAA,0D;MhK2LA,+B;MgK3LA,4B;QAQqD,OhK6LX,egK7LoB,kBhKoFvB,cgKpFuB,ChK
6LpB,C;MgK7LW,C;KARrD,C;4FAUA,yB;MAAA,wD;MhKiLA,+B;MgKjLA,4B;QAQoD,OhKmLV,egKnLmB,i
BhK0EtB,cgK1EsB,ChKmLnB,C;MgKnLU,C;KARpD,C;+EAUA,yB;MAAA,4C;MhKuKA,+B;MgKvKA,sC;QAc
2D,OhKmKjB,egKnK0B,WhK0D7B,cgK1D6B,EAAW,QAAX,ChKmK1B,C;MgKnKiB,C;KAd3D,C;iFAgBA,yB;
MtFoDA,4C;M1EmGA,+B;MgKvJA,sC;QAc4D,OhKmJIB,e0EhGuB,W1ET1B,c0ES0B,EAAW,CsFnDK,QtFmD
L,IAAX,C1EgGvB,C;MgKnJkB,C;KAd5D,C;oFAgBA,yB;MhM8CI,6B;MgM9MJ,gD;MAgKA,4B;QAO8C,OA
KO,a/LsLhB,CD0BE,cAAU,cAAL,GAAiB,GAAtB,CC1BF,M+LlLgB,C;MAGKP,C;KAP9C,C;oGASA,yB;MtFzH
A,iB;MsFyHA,4B;QAOsD,OtFzHoB,a1G0GnC,cwLde,GAAy,G9E5FQ,C8E4FrB,GAA6C,EAA7C,I;MQ6BC,C;K
APtD,C;sGASA,yB;MR7BA,kE;MQ6BA,4B;QAOuD,OR9BkB,sBxLMIC,cwLNGB,GAAW,GAAO,C;MQ8BIB,C;

KAPvD,C;8FASA,yB;MAAA,0D;MhMkFA,0B;MAAA,+B;MgMIFA,4B;QAQqD,OhMqFZ,eAAW,OgMrFS,kBh
MdnB,cAAL,GAAiB,GgMcO,ChMqFT,CAAX,C;MgMrFY,C;KARrD,C;4FAUA,yB;MAAA,wD;MhMwEA,0B;M
AAA,+B;MgMxEA,4B;QAQoD,OhM2EX,eAAW,OgM3EQ,iBhMxBiB,cAAL,GAAiB,GgMwBM,ChM2ER,CAA
X,C;MgM3EW,C;KARpD,C;gFAWA,yB;MAAA,4C;MhMsCA,+B;MgMtCA,sC;QAc2D,OhMkCjB,egMIC0B,Wh
M7D7B,cgM6D6B,EAAW,QAAX,ChMkC1B,C;MgMICiB,C;KAd3D,C;kFAgBA,yB;MAAA,8C;MhMsBA,+B;M
gMtBA,sC;QAc4D,OhMkBiB,egMIB2B,YhM7E9B,cgM6E8B,EAAZ,QAAX,ChMk3B,C;MgMIBkC,C;KAd5D,
C;oFAgBA,yB;M9LjCI,6B;M8L/MJ,gD;MAGPA,4B;QAO+C,OAhPM,a/LsLhB,CC2BE,cAAU,cAAL,GAAiB,KA
AtB,CD3BF,M+LtlG,C;MAGPN,C;KAP/C,C;oGASA,yB;MtFzMA,iB;MsFyMA,4B;QAOuD,OtFzMMB,axGqHj
C,csLiDpC,GAAY,K9EtKyD,C8EsKtE,GAA+C,EAA/C,I;MQmCmD,C;KAPvD,C;sGASA,yB;MRnCA,kE;MQm
CA,4B;QAOwD,ORpCoB,sBtlZDnC,csLyDe,GAAW,KAAS,C;MQoCpB,C;KAPxD,C;8FASA,yB;MAAA,0D;M9
LGA,4B;MAAA,iC;M8LHA,4B;QAQuD,O9LMZ,gBAAY,Q8LNQ,kB9L7FrB,cAAL,GAAiB,K8L6FS,C9LMR,C
AAZ,C;M8LNY,C;KARvD,C;4FAUA,yB;MAAA,wD;M9LPA,4B;MAAA,iC;M8LOA,4B;QAQsD,O9LJX,gBAA
Y,Q8LIO,iB9LvGpB,cAAL,GAAiB,K8LuGQ,C9LJP,CAAZ,C;M8LIW,C;KARtD,C;gFAWA,yB;MAAA,4C;M9L
7BA,iC;M8L6BA,sC;QAc6D,O9LjChB,gB8LiC0B,W9LI9B,c8LkI8B,EAAW,QAAX,C9LjC1B,C;M8LiCgB,C;K
Ad7D,C;kFAgBA,yB;MAAA,8C;M9L7CA,iC;M8L6CA,sC;QAc8D,O9LjDjB,gB8LiD2B,Y9LIJ/B,c8LkI+B,EAA
Y,QAAX,C9LjD3B,C;M8LiDiB,C;KAd9D,C;;MC9SI,ShMyHoD,cgMzH3C,ChMyH2C,EgMzHvC,ChMyHuC,C;
MgMxHpD,ShMwHoD,cgMxH3C,ChMwH2C,EgMxHvC,ChMwHuC,C;MgMvHpD,OhMiDkE,YgMjDvD,EhMiD
wE,KAAjB,EgMjDjD,EhMiD8E,KAA7B,CgMjDvD,KAAX,GhMgFsD,SgMhFjC,EhMgF2C,KAAK,GgMhF3C,E
hMgFuD,KAAZ,IAAf,CgMhFtD,GhMmEqD,SAAU,CAaT,SgMhFpB,EhMgF8B,KAAK,GgMhF9B,EhMgF0C,K
AAZ,IAAf,CabS,MAAK,GgMnExB,ChMmEmC,KAAX,IAAf,C;IgmIEzD,C;;MAGI,SjKsHsD,eiKtH7C,CjKsH6
C,EiKtHzC,CjKsHyC,C;MiKrHtD,SjKqHsD,eiKrH7C,CjKqH6C,EiKrHzC,CjKqHyC,C;MiKpHtD,OjKmDmE,aiK
nDxD,EjKmD0E,KAAiB,EiKnDiD,EjKmDgF,KAA9B,CiKnDxD,KAAX,GjK6EwD,UiK7EnC,EjK6E8C,KAAK,
UiK7E9C,EjK6E0D,KAAZ,CAAhB,CiK7ExD,GjKgEuD,UAAW,CAaV,UiK7EtB,EjK6EiC,KAAK,UiK7EjC,EjK
6E6C,KAAZ,CAAhB,CabU,MAAK,KiKhE3B,CjKgEsC,KAAZ,CAAhB,C;IiK/D3D,C;;MAqBI,WAAO,CAAP,C
;QAD8E,OhMsBZ,YgMrBiD,KhMqBmE,KAAjB,EgMrBzC,GhMqBsE,KAA7B,CgMrBiD,KAD8D,GACHD,GAD
gD,GhMqDxB,SgMpDf,GhMoDyB,KAAK,GgMpDxB,mBAAiB,GAAjB,EAAAsB,KAAtB,EhMkQV,SgMIQuC,Ih
MkQvC,CgMIQU,ChMoDoC,KAAZ,IAAf,C;;YgMnDtD,WAAO,CAAP,C;QAF8E,OhMsBZ,YgMpBiD,KhMoBm
E,KAAjB,EgMpBzC,GhMoBsE,KAA7B,CgMpBiD,KAF8D,GAehD,GAfGD,GhMwCzB,SgMtCd,GhMsCwB,KA
AK,GgMtCvB,mBAAiB,KAAjB,EAAwB,GAAxB,EhMiQV,SgMjQwC,CAAC,IAAD,IhMiQxC,CgMjQU,ChMsC
kC,KAAZ,IAAf,C;;QgMrC7C,MAAa,gCAAYB,eAAzB,C;IAHyD,C;;MAyB9E,sBAAO,CAAP,C;QADkF,OjKMf
,aiKLnD,KjKKqE,KAAiB,EiKL1C,GjKKwE,KAA9B,CiKLnD,KADkE,GACpD,GADoD,GjKgC1B,UiK/BjB,GjK
+B4B,KAAK,UiK/B3B,mBAAiB,GAAjB,EAAAsB,KAAtB,EjKyPP,UiKzPoC,IjKyPpC,CiKzPO,CjK+BuC,KAAZ,
CAAhB,C;;YiK9BxD,sBAAO,CAAP,C;QAFkF,OjKMf,aiKJnD,KjKIqE,KAAiB,EiKJ1C,GjKIwE,KAA9B,CiKJn
D,KAFkE,GAEPD,GAFOd,GjKmB3B,UiKjBhB,GjKiB2B,KAAK,KiKjB1B,mBAAiB,KAAjB,EAAwB,GAAxB,E
jKwPP,UiKxPsC,IAAD,ajKwPrC,CiKxPO,CjKiBqC,KAAZ,CAAhB,C;;QiKhB/C,MAAa,gCAAYB,eAAzB,C;IA
H6D,C;;M/L/ClF,8B;MAF0D,gB;IAAnC,C;;MAEvB,gC;MAcI,iBAGqC,WAAO,CAAP,C;MAErC,iBAGqC,WA
AO,MAAP,C;MAErC,kBAGmC,C;MAEnC,iBAGkC,E;IANbtC,C;;MAAA,uC;QAAA,sB;;MAAA,gC;IAAA,C;;
MA4B0D,OAAa,0BAkJc,SAAL,GAAiB,KAIJqB,EAAU,KFiJ3C,KAAL,GAAiB,GEJjQb,C;IAAb,C;;MASU,OA
Aa,0BAyI3C,SAAL,GAAiB,KAZI+B,EAAU,KAYIrD,KAAL,GAAiB,KAZI+B,C;IAAb,C;wGAEPe,yB;MAGKA,6
B;MDjKA,8C;MCCA,wB;QAMyD,ODAS,YAAiB,CCmKhD,cAAU,SAAL,GAAiB,KAAtB,CDnKgD,MAAJB,EC
Ae,KDAc,KAA7B,C;MCAT,C;KANzD,C;wGAQA,yB;MAkKA,aAS6D,0B;MAT7D,+B;M8BnKA,gD;M9BCA,w
B;QAM0D,O8BAS,aAAkC,C9BqKhD,eAAW,oBAAL,SAAK,CAAL,YAAN,C8BrKgD,MAAIB,E9BAGb,K8BAC,
KAA9B,C;M9BAT,C;KAN1D,C;8FAQA,yB;MAGJA,6B;MAhJA,wB;QAEsD,ODMD,cAAU,CCiJ5B,cAAU,SAAL,
GAAiB,KAAtB,CDjJ4B,MAAK,GAAW,CDgJ5C,cEtJsC,KFsJ5B,KAAL,GAAiB,GAAtB,CChJ4C,MAAX,IAAf
,C;MCNC,C;KAFtD,C;8FAGA,yB;MA6IA,6B;MA7IA,wB;QAEuD,ODGF,cAAU,CCiJ5B,cAAU,SAAL,GAAiB,
KAAtB,CDjJ4B,MAAK,GAAW,CCiJ5C,cApJuC,KAOJ7B,KAAL,GAAiB,KAAtB,CDjJ4C,MAAX,IAAf,C;MCH
E,C;KAFvD,C;8FAGA,yB;MA0IA,6B;MA1IA,wB;QAEqD,ODAA,cAAU,CCiJ5B,cAAU,SAAL,GAAiB,KAAtB,
CDjJ4B,MAAK,GCAI,KDAO,KAAZ,IAAf,C;MCAA,C;KAFrD,C;8FAGA,yB;MAiJA,aAS6D,0B;MAT7D,+B;M

AjJA,wB;QAEuD,O8BAA,eAAW,C9BwJ7B,eAAW,oBAAL,SAAK,CAAL,YAAN,C8BxJ6B,MAAK,K9BAI,K8B
AO,KAAx,CAAhB,C;M9BAA,C;KAFvD,C;gGAIA,yB;MAmIA,6B;MAnIA,wB;QAEuD,ODMD,cAAU,CCoI7B,
cAAU,SAAL,GAAiB,KAAtB,CDpI6B,MAAK,GAAy,CDmI9C,cEzIwC,KFyI9B,KAAL,GAAiB,GAAtB,CCnI8C,
MAAZ,IAAf,C;MCNC,C;KAFvD,C;gGAGA,yB;MAgIA,6B;MAhIA,wB;QAEwD,ODGF,cAAU,CCoI7B,cAAU,S
AAL,GAAiB,KAAtB,CDpI6B,MAAK,GAAy,CCoI9C,cAvIyC,KAuI/B,KAAL,GAAiB,KAAtB,CDpI8C,MAAZ,I
AAf,C;MCHE,C;KAFxD,C;gGAGA,yB;MA6HA,6B;MA7HA,wB;QAEsD,ODAA,cAAU,CCoI7B,cAAU,SAAL,G
AAiB,KAAtB,CDpI6B,MAAK,GCAK,KDAO,KAAZ,IAAf,C;MCAA,C;KAFtD,C;gGAGA,yB;MAoIA,aAS6D,0B
;MAT7D,+B;MApIA,wB;QAEwD,O8BAA,eAAW,C9B2I9B,eAAW,oBAAL,SAAK,CAAL,YAAN,C8B3I8B,MA
AK,U9BAK,K8BAO,KAAZ,CAAhB,C;M9BAA,C;KAFxD,C;gGAIA,yB;MAshA,6B;MAthA,wB;QAEuD,ODM
D,cAAe,YAAL,CCuH7B,cAAU,SAAL,GAAiB,KAAtB,CDvH6B,MAAK,EAAY,CDsH9C,cE5HwC,KF4H9B,KA
AL,GAAiB,GAAtB,CCtH8C,MAAZ,CAAf,C;MCNC,C;KAFvD,C;gGAGA,yB;MAmHA,6B;MAnHA,wB;QAEw
D,ODGF,cAAe,YAAL,CCuH7B,cAAU,SAAL,GAAiB,KAAtB,CDvH6B,MAAK,EAAY,CCuH9C,cA1HyC,KA0H
/B,KAAL,GAAiB,KAAtB,CDvH8C,MAAZ,CAAf,C;MCHE,C;KAFxD,C;gGAGA,yB;MAgHA,6B;MAhHA,wB;
QAEsD,ODAA,cAAe,YAAL,CCuH7B,cAAU,SAAL,GAAiB,KAAtB,CDvH6B,MAAK,ECAK,KDAO,KAAZ,CA
Af,C;MCAA,C;KAFtD,C;gGAGA,yB;MAuHA,aAS6D,0B;MAT7D,+B;MAvHA,wB;QAEwD,O8BAA,eAAW,C9
B8H9B,eAAW,oBAAL,SAAK,CAAL,YAAN,C8B9H8B,MAAK,U9BAK,K8BAO,KAAZ,CAAhB,C;M9BAA,C;
KAFxD,C;4FAIA,yB;MAyGA,6B;MDnGA,4C;MCNA,wB;QAEqD,ODMD,WC0GjB,cAAU,SAAL,GAAiB,KAAt
B,CD1GiB,EDyGjB,cE/GoC,KF+G1B,KAAL,GAAiB,GAAtB,CCzGiB,C;MCNC,C;KAFrD,C;4FAGA,yB;MAsh
A,6B;MDnGA,4C;MCHA,wB;QAEsD,ODGF,WC0GjB,cAAU,SAAL,GAAiB,KAAtB,CD1GiB,EC0GjB,cA7GqC,
KA6G3B,KAAL,GAAiB,KAAtB,CD1GiB,C;MCHE,C;KAFtD,C;4FAGA,yB;MAmGA,6B;MDnGA,4C;MCAA,w
B;QAEoD,ODAA,WC0GjB,cAAU,SAAL,GAAiB,KAAtB,CD1GiB,ECAkB,KDAIB,C;MCAA,C;KAFpD,C;4FAG
A,yB;MA0GA,aAS6D,0B;MAT7D,+B;M8B1GA,8C;M9BAA,wB;QAEsD,O8BAA,Y9BiHjB,eAAW,oBAAL,SA
AK,CAAL,YAAN,C8BjHiB,E9BAmB,K8BAnB,C;M9BAA,C;KAFtD,C;4FAIA,yB;MA4FA,6B;MDtFA,kD;MCN
A,wB;QAEqD,ODMD,cC6FjB,cAAU,SAAL,GAAiB,KAAtB,CD7FiB,ED4FjB,cEIgoC,KfKGI1B,KAAL,GAAiB,
GAAtB,CC5FiB,C;MCNC,C;KAFrD,C;4FAGA,yB;MAyFA,6B;MDtFA,kD;MCHA,wB;QAEsD,ODGF,cC6FjB,c
AAU,SAAL,GAAiB,KAAtB,CD7FiB,EC6FjB,cAhGqC,KAgG3B,KAAL,GAAiB,KAAtB,CD7FiB,C;MCHE,C;K
AFtD,C;4FAGA,yB;MAshA,6B;MDtFA,kD;MCAA,wB;QAEoD,ODAA,cC6FjB,cAAU,SAAL,GAAiB,KAAtB,C
D7FiB,ECAkB,KDAIB,C;MCAA,C;KAFpD,C;4FAGA,yB;MA6FA,aAS6D,0B;MAT7D,+B;M8B7FA,oD;M9BAA
,wB;QAEsD,O8BAA,e9BoGjB,eAAW,oBAAL,SAAK,CAAL,YAAN,C8BpGiB,E9BAmB,K8BAnB,C;M9BAA,C;
KAFtD,C;4EAIA,yB;MAAA,4B;MAAA,iC;MAAA,mB;QAE2C,uBAAY,QAAL,SAAK,KAAZ,C;MAAA,C;KAF
3C,C;4EAGA,yB;MAAA,4B;MAAA,iC;MAAA,mB;QAE2C,uBAAY,QAAL,SAAK,KAAZ,C;MAAA,C;KAF3C,
C;oGAIA,yB;MAAA,8C;MAwEA,6B;MAxEA,wB;QAE+D,0BA+E5B,cAAU,SAAL,GAAiB,KAAtB,CA/E4B,EA
+E5B,cA/EqD,KA+E3C,KAAL,GAAiB,KAAtB,CA/E4B,C;MAAA,C;KAF/D,C;4FAIA,yB;MAAA,iC;M8JvHJ,4
B;M9JuHI,wB;QAEqD,uB8JtHiC,Q9JsH1B,IAAK,K8JtHX,G9JsHoB,KAAM,K8JtHM,C9JsHjC,C;MAAA,C;KAF
rD,C;0FAGA,yB;MAAA,iC;M8JrHJ,4B;M9JqHI,wB;QAEoD,uB8JpHgC,Q9JoHzB,IAAK,K8JpHX,G9JoHmB,K
AAM,K8JpHM,C9JoHhC,C;MAAA,C;KAFpD,C;4FAGA,yB;MAAA,iC;M8JnHJ,4B;M9JmHI,wB;QAEqD,uB8Jl
HiC,Q9JkH1B,IAAK,K8JIHX,G9JkHoB,KAAM,K8JIHM,C9JkHjC,C;MAAA,C;KAFrD,C;4EAGA,yB;MAAA,iC;
M8JjHJ,4B;M9JiHI,mB;QAEkC,uB8JhHsB,QAAP,C9JgHR,S8JhHe,C9JgHtB,C;MAAA,C;KAFIC,C;kFAIA,yB;M
AAA,0B;MAAA,mB;QAUmC,OAAK,OAAL,SAAK,C;MAAL,C;KAVnC,C;MAoBqC,gB;IAAA,C;MAUJ,OAA
K,SAAL,GAAiB,K;IAAjB,C;kFACjC,yB;MAAA,aASqD,0B;MATrD,mB;QASmC,OAAK,oBAAL,SAAK,CAAL,
Y;MAAA,C;KATnC,C;oFAWA,yB;MF+DJ,0B;MAAA,+B;ME/DI,mB;QASqC,OFiEE,eAAW,OEjEb,SFiEa,CAA
X,C;MEjEF,C;KATrC,C;MAYuC,W;IAAA,C;kFACvC,yB;MAAA,6B;MAAA,mB;QASmC,qBAAU,SAAL,GAA
iB,KAAtB,C;MAAA,C;KATnC,C;oFAUA,yB;MAAA,aAS6D,0B;MAT7D,+B;MAAA,mB;QASqC,sBAAW,oBA
AL,SAAK,CAAL,YAAN,C;MAAA,C;KATrC,C;MAiBqC,OApDC,SAAL,GAAiB,K;IAoDb,C;MAOE,OA3DD,
SAAL,GAAiB,K;IA2DX,C;MAEE,OAAQ,CA7DX,SAAL,GAAiB,KA6DD,Y;IAAR,C;MAhP7C,c;MAG8D,q
D;MAH9D,a;IAAA,C;MAAA,2IAG8D,oCAH9D,G;IAAA,C;0EAoPA,yB;MAAA,iC;MAAA,4B;QAW4C,uBAA
Y,SAAZ,C;MAAA,C;KAX5C,C;4EAYA,yB;MAAA,iC;MAAA,4B;QAU6C,uBAAO,SAAP,C;MAAA,C;KAV7C,
C;4EAWA,yB;MAAA,4B;MAAA,iC;MAAA,4B;QAW2C,uBAAY,QAAL,SAAK,CAAZ,C;MAAA,C;KAX3C,C;4

EAYA,yB;MAAA,4B;MAAA,iC;MAAA,4B;QAW4C,uBAAY,QAAL,SAAK,SAAZ,C;MAAA,C;KAX5C,C;;M+
BpRqB,sB;IAFrB,C;;MAAkD,O/B4PL,W+B5PK,aAAQ,KAAR,C/B4PL,C;I+B5PK,C;;MAS1C,aAAQ,KAAR,IAAI
B,K/BkJgB,K;I+BjJrC,C;;MAGsC,OAAA,YAAQ,O;IAAR,C;;MAGoB,gCAAS,YAAT,C;IAAA,C;;MAEF,yB;MA
AjC,oB;MACnB,eAAoB,C;IADF,C;;MAEO,sBAAQ,YAAM,O;IAAd,C;;MACiC,Q;MAA9B,IAAI,eAAQ,YAAM,
OAAIB,C;QAAA,O/BuOS,W+BvOe,aAAM,mBAAN,EAAM,2BAAN,O/BuOf,C;;;Q+BvO8C,MAAM,2BAAuB,Y
AAM,WAA7B,C;IAA7D,C;;;MAMvB,Q;MAAL,IAAI,eAAC,0EAAD,SAAJ,C;QAAkC,OAAO,K;MAEZC,OAAe,
WAAR,YAAQ,EAAS,O/B8HS,K+B9HIB,C;IACnB,C;;MAGY,Q;MAA2B,gBAA3B,gE;MAA2B,c;;QhBw4CvB,
U;QADhB,IAAI,wCAAsB,mBAA1B,C;UAAqC,aAAO,I;UAAP,e;;QACrB,6B;QAAhB,OAAgB,gBAAhB,C;UAA
gB,2B;UgBx4C6B,2BhBw4CR,OgBx4CQ,S;UAAA,W;YAAwB,oBAAR,YAAQ,EhBw4ChC,Of9wCA,K+B1HgC,
C;;UhBw4C/C,IAAI,OAAJ,C;YAAyB,aAAO,K;YAAP,e;;;QAC/C,aAAO,I;;MgBz4CH,iB;IACJ,C;;MAEKc,OAA
A,IAAK,QAAQ,OAAb,KAAqB,C;IAArB,C;;MA7CIC,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADhC,Y;I
AAA,C;;;MAPJ,OAKqB,sDALrB,M;IAAA,C;;MAAA,c;MAKqB,wD;MALrB,a;IAAA,C;;MAAA,2IAKqB,0CA
LrB,G;IAAA,C;kFAuDA,yB;MAAA,2C;;QAWwC,wB;UAAW,OAAA,aAAK,KAAL,C/ByGV,K;Q+BzGD,C;MA
AA,C;MAXxC,6B;QAWI,OAAO,qBAAY,gCAAW,IAAX,GAAiB,wBAAjB,CAAZ,C;MACX,C;KAZA,C;;MAiB
wE,e;IAAA,C;;MiKhEb,OAAa,WAAb,SIM4KjB,KAAL,GAAiB,GkM5KkB,EAAS,KAAT,C;IAAb,C;;MAUC,OA
Aa,WAAb,ShMmKIB,KAAL,GAAiB,KgMnKmB,EAAS,KAAT,C;IAAb,C;;MAWF,OAAc,WjMyK5B,oBiMzKc,
SjMyKnB,KAAC,CAAL,iBiMzKiC,EAAS,KAAT,C;IAAd,C;;MASV,uBAAc,SIKiKvB,KkKjKS,EAA6B,WAAW,
KAAAX,CAA7B,C;IAAA,C;;MASX,Q;MAAA,0DAAmB,kBAAkB,SAAlB,C;IAAnB,C;;MASU,Q;MAAA,0CAAc,
KAAAd,oBAAwB,kBAAkB,SAAlB,C;IAAxB,C;;MASR,Q;MAAA,2DAAoB,kBAAkB,SAAlB,C;IAApB,C;;MASU
,Q;MAAA,2CAAE,KAAf,oBAAYB,kBAAkB,SAAlB,C;IAAZB,C;;MAQd,Q;MAAA,yDAAkB,kBAAkB,SAAlB,C
;IAAlB,C;;MASU,Q;MAAA,yCAAA,KAAb,oBAAuB,kBAAkB,SAAlB,C;IAAvB,C;;MAQR,Q;MAAA,0DAAmB,
kBAAkB,SAAlB,C;IAAnB,C;;MASU,Q;MAAA,0CAAc,KAAAd,oBAAwB,kBAAkB,SAAlB,C;IAAxB,C;;MAYH,
kCAAsB,EAAtB,C;IAAA,C;;MAWzB,Q;MAAA,sBAAL,SAAK,EAAa,KAAb,C;MAAL,iB;QAA4B,OAAO,I;;M
AA7C,UAAU,I;MACV,IjMhFkE,YiMgF9D,GjMhF+E,KAAjB,EAA6B,CDkK5D,SkMIFzB,6BAAM,UIMkF6B,K
AAL,GAAiB,GAAtB,CCIK4D,MAA7B,CiMgF9D,IAAJ,C;QAA2B,OAAO,I;MACiC,OIMkJqC,UAAW,OkMIJzC
,GjM4E8B,KDsEW,CAAX,C;IkMjJzC,C;;MAQ8C,mCAAuB,EAAvB,C;IAAA,C;;MAW3B,Q;MAAA,sBAAL,SA
AK,EAAa,KAAb,C;MAAL,iB;QAA4B,OAAO,I;;MAA7C,UAAU,I;MACV,IjMtGkE,YiMsG9D,GjMtG+E,KAAj
B,EAA6B,CCmK5D,SgM7DzB,8BAAO,UhM6D4B,KAAL,GAAiB,KAAtB,CDnK4D,MAA7B,CiMsG9D,IAAJ,C
;QAA4B,OAAO,I;MACnB,OhM6HuC,WAAy,QgM7H5C,GjMgEgC,KC6DY,CAAZ,C;Igm5H3C,C;;MAQ0C,iC
AAqB,EAARb,C;IAAA,C;;MAWtC,WAAW,KAAAX,C;MAEA,aAAa,SAAK,O;MACiB,IAAI,WAAU,CAAd,C;QA
AiB,OAAO,I;MAExB,YAAkB,4BAAK,U;MACvB,S;MAEA,gBAAGb,qBAAK,CAAL,C;MACHb,IAAI,YAAY,EA
AhB,C;QACI,IAAI,WAAU,CAAV,IAAe,cAAa,EAAhC,C;UAAqC,OAAO,I;QAC5C,QAAQ,C;;QAER,QAAQ,
C;;MAGZ,ajMkGmC,SiMIGtB,KjMkGsB,C;MiMjGnC,qBjMnFoD,WiMmF/B,KjMnF+B,EiMmFvB,MjMnFuB,C;
MiMoFpD,aAAa,W;MACb,aAAU,KAAV,MAAsB,MAAtB,M;QACI,YAAY,QAAQ,qBAAK,CAAL,CAAR,EAAi
B,KAAjB,C;QAEZ,IAAI,QAAQ,CAAZ,C;UAAe,OAAO,I;QACtB,IjMIJ8D,YiMkJ1D,MjMIJ2E,KAAjB,EiMkJjD,
cjMIJ8E,KAA7B,CiMkJ1D,IAAJ,C;UAA6B,OAAO,I;QAEpC,SjMxGkD,SAAE,YiMwGjE,MjMxG4D,KAAK,Ei
MwGvD,MjMxGmE,KAAZ,CAAF,C;QiM0GID,mBAAmB,M;QACnB,SjMrliD,SiMqIjD,MjMrI2D,KAAK,GAA
W,CA2N5C,SiMtFrB,KjMsFqB,CA3N4C,MAAX,IAAF,C;QiMsIjD,IjMxJ8D,YiMwJ1D,MjMxJ2E,KAAjB,EiMwJ
jD,YjMxJ8E,KAA7B,CiMwJ1D,IAAJ,C;UAA2B,OAAO,I;;MAGtC,OAAO,M;IACX,C;;MAQ4C,kCAAsB,EAAtB
,C;IAAA,C;;MAWxC,WAAW,KAAAX,C;MAEA,aAAa,SAAK,O;MACiB,IAAI,WAAU,CAAd,C;QAAiB,OAAO,I;
MAExB,YAAmB,6BAAM,U;MACzB,S;MAEA,gBAAGb,qBAAK,CAAL,C;MACHb,IAAI,YAAY,EAAhB,C;QA
CI,IAAI,WAAU,CAAV,IAAe,cAAa,EAAhC,C;UAAqC,OAAO,I;QAC5C,QAAQ,C;;QAER,QAAQ,C;;MAIZ,ajM
6CmC,SiM7CtB,KjM6CsB,C;MiM5CnC,qBkRiSd,YkKqIjC,KIKrliC,E/BwHjB,UAAW,oBiManB,MjMbc,KAAK,
CAAL,iBAAN,C+BxHiB,C;MkKsItD,aAAa,2B;MACb,aAAU,KAAV,MAAsB,MAAtB,M;QACI,YAAY,QAAQ,q
BAAK,CAAL,CAAR,EAAiB,KAAjB,C;QAEZ,IAAI,QAAQ,CAAZ,C;UAAe,OAAO,I;QACtB,IiK/L+D,akK+L3D
,MIK/L6E,KAAiB,EkK+L1D,clK/LgF,KAA9B,CkK+L3D,IAAJ,C;UAA6B,OAAO,I;QAEpC,SiK1JoD,UkK0JpD,
MIK1J+D,KAAK,UAAy,C/BqI/C,UAAW,oBiMqBIC,MjMrB6B,KAAK,CAAL,iBAAN,C+BrI+C,MAAZ,CAAhB
,C;QkK4JpD,mBAAmB,M;QACnB,SIKvLmD,UkKuLnD,MIKvL8D,KAAK,KAAW,C/B+J7C,UAAW,oBAAL,C

KAAa,GAAjB,GAAsB,mCAAtB,GAaID,I;IAAJD,C;;;;;MCzD0C,wB;IAAD,C;;MAAC,yB;IAAA,C;;;MAEtD,q
C;IAAA,C;;MAK6D,W;IAAA,C;;MACmB,c;IAAA,C;;MACJ,c;IAAA,C;;MACV,W;IAAA,C;;MACxB,Q;IAAA,C
;;MACG,8B;IAAA,C;;;MAV7C,4C;QAAA,2B;;MAAA,qC;IAAA,C;;MAiB+B,gB;MAA4B,sB;IAA7B,C;;MAIIB,
Q;MAFJ,UAAU,I;MACV,OAAO,IAAP,C;QACI,YAAA,GAAI,QAaJ,aAAY,GAaZ,W;UAAwB,W;;QACxB,WA
AW,GAaI,K;QACf,IAAI,oCAAJ,C;UACI,MAAM,I;;UAEN,OAAO,iBAaK,GAAL,C;;;IAGnB,C;;MAGI,iBAaU
,SAaK,cAAK,OAAL,EAaC,SAaD,CAaF,EAaYc,YAAzC,C;IAAA,C;;UAKO,I;MAFP,6BAaQ,GAAR,U;QAAo
B,OAAO,S;;MAC3B,cAAc,SAaK,kBAAS,GAAT,C;MAEf,gBAAY,SAaZ,C;QAAoB,W;WACpB,gBAAY,mCA
AZ,C;QAAqC,mB;;QAC7B,2BAaG,B,OAaH,B,EAaYB,YAAzB,C;MAHZ,W;IAKJ,C;;MAGI,OAaI,yCAAJ,GAA
6B,SAaK,SAAL,GAaC,CAaD,IAA7B,GAaK,D,C;IAAID,C;;MAGA,+BAaI,OAAQ,IAAZ,GAAoB,OAApB,C;IA
AA,C;;MAUwB,Q;MAPxB,UAAU,O;MACV,OAAO,IAAP,C;QACI,IAAI,CAAC,gBAAS,GAaI,QAaB,CAAL,C;
UAA4B,OAAO,K;QACnC,WAAW,GAaI,K;QACf,IAAI,oCAAJ,C;UACI,MAAM,I;;UAEN,OAAO,gBAAS,wEA
AT,C;;;IAGnB,C;;MAGI,gBAAS,KAAT,KAaK,B,yCAA4B,KAAM,SAAN,KAaG,B,aAA5C,IAAsD,KAAM,eAAY
,IAAZ,CAA9E,C;IAAA,C;;MAE2B,OAaK,SAAL,SAaK,CAAL,GAA0B,SAAR,YAAQ,CAA1B,I;IAAA,C;;MAI
vB,OAaI,G9KJoC,YAAU,C8KpJID,GAaM,B,OAAQ,WAA3B,GAA2C,MAAM,IAAN,YAAa,OAaB,C;IAC/C,C;
;MAFA,aAAM,kBAaK,EAAL,EAAS,+BAAT,CAAN,GAeI,G;IAFJ,C;;;;;ICvDR,C;;;MCI0C,uB;QAAU,qBAAM
,IAaK,OAaX,C;MAAV,C;IAAA,C;;MNgDb,oB;QACrB,WAAW,sBAaI,B,CAAjB,C;QACX,cAAM,IAAN,C;QA
DA,OAeA,IAaK,Y;MAHgB,C;IAAA,C;;MAvDzB,oCAAYB,QAaZB,EAaM,C,UAAAnC,CAA+C,cAAO,IAAP,C;I
ACnD,C;;MAYI,sCAAYB,UAAzB,CAAqC,cAAO,IAAP,C;IACzC,C;;MAewB,4BAaI,B,oCAAYB,QAaZB,EAaM
C,UAAAnC,CAAjB,EAaI,e,mBAaJ,E,C;IAAA,C;;MAcA,4BAaI,B,sCAAYB,UAAzB,CAAjB,EAaU,D,mBAaV,D,C;I
AAA,C;;MAYpB,OMhDsC,gCNgDb,8BMhDa,CAaT,C,c;INgDA,C;2FAVJ,yB;MAAA,0F;;QMtC0C,uB;UAAU,q
BAAM,IAaK,OAaX,C;QAAV,C;MAAA,C;;QNgDb,oB;UACrB,WAAW,sBAaI,B,CAAjB,C;UACX,cAAM,IAA
N,C;UADA,OAeA,IAaK,Y;QAHgB,C;MAAA,C;MAV7B,sC;QMtCI,mBAAsC,gCNgDb,8BMhDa,CAaT,C,6B;Q
NgDA,yD;MAAA,C;KAVJ,C;kHA4BI,yB;MAAA,gE;MAAA,mB;QACI,MAAM,6BAaO,B,0BAaP,B,C;MACV,C;
KAFA,C;4IAMJ,yB;MAAA,yF;MAAA,gE;MAAA,8B;MAAA,qB;MAAA,oC;QAMa,Q;;UAHL,aAAa,O;UACb,I
AAI,WAAW,mBAaF,C;YAEuC,CAaIC,kEAaK,C,eAAO,MAAP,C;;;UAEzC,gC;YACE,UAAW,6BAaO,B,CAaP
B,C;;YAPf,O;;MASJ,C;KAXA,C;;MMxE0C,uB;QAAU,qBAAM,IAaK,OAaX,C;MAAV,C;IAAA,C;;MAaT,C,O
AAsC,wCAaT,C,c;IAAA,C;8GARBJ,yB;;QAqB0C,uB;UAAU,qBAAM,IAaK,OAaX,C;QAAV,C;MAAA,C;MARB
1C,sC;QAqBI,mBAAsC,sCAaT,C,6B;QAAA,yD;MAAA,C;KArBJ,C;;MAiCI,MAAM,wBAaO,B,sEAaP,B,C;IAAN
,C;2HAVJ,yB;MAAA,gE;MAAA,sC;QAUI,MAAM,6BAaO,B,sEAaP,B,C;MAAN,C;KAVJ,C;mHAYa,yB;MAAA
,gE;MAAA,4B;QAMI,MAAM,6BAaO,B,4CAaP,B,C;MAAN,C;KANJ,C;;MjM7BuF,wC;IARvF,C;IAAA,wD;MA
S2C,8B;IAAA,C;IAT3C,8E;;MiMuB0C,uB;QAAU,qBAAM,IAaK,OAaX,C;MAAV,C;IAAA,C;;MCVmF,mB;Q
AAE,qBAaC,qBAaD,C;MAAF,C;IAAA,C;;MAAT,OIML7B,2BkMKsC,mCIMLtC,C;IkMK6B,C;;MAWhH,eAAe,
6B;MACf,oBAaK,C,yBAaD,aAAc,EAaO,C,QAaP,C,EAa2D,QAa3D,C;MACiC,OAAO,Q;IACX,C;;IAa4C,C;;MA
iBpC,IAAI,uCAA0B,QAAS,UAAvC,C;QAaK,D,M;MACiD,OAAO,sBAAS,QAAS,WAAI,B,e;IACX,C;;MAOqD,6
BAAS,QAAS,WAAI,B,e;IAAA,C;;;;;MAYd,0B;MACvC,eAAoB,C;MACpB,mBAa4B,I;MAC5B,sBAAYC,I;M
ACzC,gBAaO,C,I;IAJxC,C;;MAOQ,OAAO,IAAP,C;QACI,QAAM,YAAN,C;eACI,C;YAAA,K;eACA,C;YACI,IA
AI,kCAaE,UAAAnB,C;cACI,eAAQ,C;cACR,OAAO,I;;cAEP,sBAaE,I;;YALvB,K;eAOA,C;YAAc,OAAO,K;eACr
B,C;eAAA,C;YAAgC,OAAO,I;kBAC/B,MAAM,yB;;QAGIB,eAAQ,C;QACR,WAAW,4B;QACX,gBAAW,I;QAC
X,IAaK,cAAO,IAAP,C;;IAEb,C;;MAGU,IASe,I;MATrB,QAAM,YAAN,C;aACI,C;aAAA,C;UAAAsC,OAAO,qB;a
AC7C,C;UACI,eAAQ,C;UACR,OAAO,kCAaE,O;aAE1B,C;UACI,eAAQ,C;UACR,aCa,iF;UACb,mBAAY,I;U
ACZ,OAAO,M;gBAEH,MAAM,yB;;IAEtB,C;;MAGI,IAAI,CAAC,cAAL,C;QAaG,B,MAAM,6B;;QAA8B,OAAO
,W;IAC/D,C;;MAE4C,QAAM,YAAN,C;aACxC,C;UADwC,OAC1B,6B;aACd,C;UAFwC,OAExB,2BAAsB,sBAa
tB,C;gBAFwB,OAGhC,2BAAsB,uCAAoC,YAAID,C;;IAHgC,C;;MAUR,oB;QAC5B,wCAAW,C;QAAX,OACA,
mB;MAF4B,C;IAAA,C;;MAFhC,mBAAY,K;MACZ,eAAQ,C;MACR,ODpHkC,kCCoHF,0CDpHE,CAaT,C,c;ICw
HA,C;;MAMoC,oB;QAC5B,wCAAW,C;QAAX,OACA,mB;MAF4B,C;IAAA,C;;MAHhC,IAAI,CAAC,QAAS,UA
Ad,C;QAAYB,M;MACzB,sBAaE,Q;MACf,eAAQ,C;MACR,OD9HkC,kCC8HF,6CD9HE,CAaT,C,c;ICKIA,C;;MAI
I,eAAQ,C;IACZ,C;;MAGI,MAAM,S;IACV,C;;MAGY,0C;IAAA,C;;;;;gBRxIc,S;cACF,
S;0BEGiB,sC;qBMsCH,C;yBACI,C;sBACH,C;kBACJ,C;iBACD,C;mBACE,C;;;A3NvExC,CAJC,E;" }

Found in path(s):

* /opt/cola/permits/1411867106_1662683866.8338356/0/kotlin-1-3-50-4-tgz/package/kotlin.js.map

1.190 kotlinox-coroutines-core 1.1.1

1.190.1 Available under license :

```
=====
== NOTICE file corresponding to the section 4 d of           ==
== the Apache License, Version 2.0,                          ==
== in this case for the kotlix.coroutines library.           ==
=====
```

kotlinox.coroutines library.

Copyright 2016-2018 JetBrains s.r.o and respective authors and developers

Copyright 2016-2018 JetBrains s.r.o.

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

The MIT License (MIT)

Copyright (c) 2016 Parker Moore

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.191 gomega v1.16.0

1.191.1 Available under license :

Copyright (c) 2013-2014 Onsi Fakhouri

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014 Amit Kumar Gupta

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.192 guava-listenablefuture-only 9999.0-empty-to-avoid-conflict-with-guava

1.192.1 Available under license :

Found license 'GNU Lesser General Public License' in '// This library is free software; you can redistribute it and/or // modify it under the terms of the GNU Lesser General Public // License as published by the Free Software Foundation; either // version 2.1 of the License, or (at your option) any later version. // This library is distributed in the hope that it will be useful, // but WITHOUT ANY WARRANTY; without even the implied warranty of // MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU // Lesser General Public License for more details. // You should have received a copy of the GNU Lesser General Public * This grammar is in the PUBLIC DOMAIN'

1.193 commons-logging 1.2

1.193.1 Available under license :

Apache Commons Logging
Copyright 2003-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.194 mbknor-jackson-jsonschema 1.0.39

1.194.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 NextGenTel

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.195 pegasus-kv/thrift v0.13.0

1.195.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get

it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with

the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the

Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the

Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time.

Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest

possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!

This package was debianized by Thrift Developer's <dev@thrift.apache.org>.

This package and the Debian packaging is licensed under the Apache License, see `~/usr/share/common-licenses/Apache-2.0`.

The following information was copied from Apache Thrift LICENSE file.

SOFTWARE DISTRIBUTED WITH THRIFT:

The Apache Thrift software includes a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

Portions of the following files are licensed under the MIT License:

lib/erl/src/Makefile.am

Please see doc/otp-base-license.txt for the full terms of this license.

The following files contain some portions of code contributed under the Thrift Software License (see doc/old-thrift-license.txt), and relicensed under the Apache 2.0 License:

compiler/cpp/Makefile.am
compiler/cpp/src/generate/t_cpp_generator.cc
compiler/cpp/src/generate/t_csharp_generator.cc
compiler/cpp/src/generate/t_erl_generator.cc
compiler/cpp/src/generate/t_hs_generator.cc
compiler/cpp/src/generate/t_java_generator.cc
compiler/cpp/src/generate/t_ocaml_generator.cc
compiler/cpp/src/generate/t_perl_generator.cc
compiler/cpp/src/generate/t_php_generator.cc
compiler/cpp/src/generate/t_py_generator.cc
compiler/cpp/src/generate/t_rb_generator.cc
compiler/cpp/src/generate/t_st_generator.cc
compiler/cpp/src/generate/t_xsd_generator.cc
compiler/cpp/src/main.cc
compiler/cpp/src/parse/t_field.h
compiler/cpp/src/parse/t_program.h
compiler/cpp/src/platform.h
compiler/cpp/src/thrift.ll
compiler/cpp/src/thrift.yy
lib/csharp/src/Protocol/TBinaryProtocol.cs
lib/csharp/src/Protocol/TField.cs
lib/csharp/src/Protocol/TList.cs
lib/csharp/src/Protocol/TMap.cs
lib/csharp/src/Protocol/TMessage.cs
lib/csharp/src/Protocol/TMessageType.cs
lib/csharp/src/Protocol/TProtocol.cs
lib/csharp/src/Protocol/TProtocolException.cs

lib/csharp/src/Protocol/TProtocolFactory.cs
lib/csharp/src/Protocol/TProtocolUtil.cs
lib/csharp/src/Protocol/TSet.cs
lib/csharp/src/Protocol/TStruct.cs
lib/csharp/src/Protocol/TType.cs
lib/csharp/src/Server/TServer.cs
lib/csharp/src/Server/TSimpleServer.cs
lib/csharp/src/Server/TThreadPoolServer.cs
lib/csharp/src/TApplicationException.cs
lib/csharp/src/Thrift.csproj
lib/csharp/src/Thrift.sln
lib/csharp/src/TProcessor.cs
lib/csharp/src/Transport/TServerSocket.cs
lib/csharp/src/Transport/TServerTransport.cs
lib/csharp/src/Transport/TSocket.cs
lib/csharp/src/Transport/TStreamTransport.cs
lib/csharp/src/Transport/TTransport.cs
lib/csharp/src/Transport/TTransportException.cs
lib/csharp/src/Transport/TTransportFactory.cs
lib/csharp/ThriftMSBuildTask/Properties/AssemblyInfo.cs
lib/csharp/ThriftMSBuildTask/ThriftBuild.cs
lib/csharp/ThriftMSBuildTask/ThriftMSBuildTask.csproj
lib/rb/lib/thrift.rb
lib/st/README
lib/st/thrift.st
test/OptionalRequiredTest.cpp
test/OptionalRequiredTest.thrift
test/ThriftTest.thrift

For the aclocal/ax_boost_base.m4 and contrib/fb303/aclocal/ax_boost_base.m4 components:

```
# Copyright (c) 2007 Thomas Porschberg <thomas@randspringer.de>
#
# Copying and distribution of this file, with or without
# modification, are permitted in any medium without royalty provided
# the copyright notice and this notice are preserved.
```

For the compiler/cpp/src/md5.[ch] components:

```
/*
Copyright (C) 1999, 2000, 2002 Aladdin Enterprises. All rights reserved.
```

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

L. Peter Deutsch
ghost@aladdin.com

*/

For the lib/rb/setup.rb: Copyright (c) 2000-2005 Minero Aoki,
lib/ocaml/OCamlMakefile and lib/ocaml/README-OCamlMakefile components:
Copyright (C) 1999 - 2007 Markus Mottl

Licensed under the terms of the GNU Lesser General Public License 2.1
(see doc/lgpl-2.1.txt for the full terms of this license)
Tue Oct 24 12:28:44 CDT 2006

Copyright (c) <2006> <Martin J. Logan, Erlware>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software (OTP Base, fslib, G.A.S) and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file

distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Thrift

Copyright (C) 2006 - 2019, The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

SOFTWARE DISTRIBUTED WITH THRIFT:

The Apache Thrift software includes a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

Portions of the following files are licensed under the MIT License:

lib/erl/src/Makefile.am

Please see doc/otp-base-license.txt for the full terms of this license.

For the aclocal/ax_boost_base.m4 and contrib/fb303/aclocal/ax_boost_base.m4 components:

```
# Copyright (c) 2007 Thomas Porschberg <thomas@randspringer.de>
#
# Copying and distribution of this file, with or without
# modification, are permitted in any medium without royalty provided
# the copyright notice and this notice are preserved.
```

For the lib/nodejs/lib/thrift/json_parse.js:

```
/*
  json_parse.js
  2015-05-02
  Public Domain.
  NO WARRANTY EXPRESSED OR IMPLIED. USE AT YOUR OWN RISK.

*/
(By Douglas Crockford <douglas@crockford.com>)
```

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.196 lz4-and-xxhash 1.7.1

1.196.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.197 aws-java-sdk-for-aws-sts 1.12.29

1.197.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.198 animal-sniffer-annotation 1.19

1.198.1 Available under license :

No license file was found, but licenses were detected in source scan.

The MIT License

Copyright (c) 2009 codehaus.org.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Found in path(s):

* /opt/cola/permits/1127519953_1611825926.02/0/animal-sniffer-annotations-1-19-sources-jar/META-
INF/maven/org.codehaus.mojo/animal-sniffer-annotations/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* The MIT License

*

* Copyright (c) 2008 Kohsuke Kawaguchi and codehaus.org.
*
* Permission is hereby granted, free of charge, to any person obtaining a copy
* of this software and associated documentation files (the "Software"), to deal
* in the Software without restriction, including without limitation the rights
* to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
* copies of the Software, and to permit persons to whom the Software is
* furnished to do so, subject to the following conditions:
*
* The above copyright notice and this permission notice shall be included in
* all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
* THE SOFTWARE.
*
*/

Found in path(s):

* /opt/cola/permits/1127519953_1611825926.02/0/animal-sniffer-annotations-1-19-sources-jar/org/codehaus/mojo/animal_sniffer/IgnoreJRERequirement.java

1.199 x-net 20210928-snapshot-e81a3d93

1.199.1 Available under license :

This source code was written by the Go contributors.
The master list of contributors is in the main Go distribution,
visible at <http://tip.golang.org/CONTRIBUTORS>.
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.200 google-android-annotations-library

4.1.1.4

1.200.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2012 The Android Open Source Project
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/ws_local/PERMITS_SQL/1014842714_1591897072.99/0/annotations-4-1-1-4-sources-
jar/android/annotation/SuppressWarnings.java
* /opt/ws_local/PERMITS_SQL/1014842714_1591897072.99/0/annotations-4-1-1-4-sources-
jar/android/annotation/TargetApi.java
```

1.201 jackson-databind 2.12.3

1.201.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.202 aws-java-sdk-::-core-::-protocols-::- json-utils 2.17.52

1.202.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software.

Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by
Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.203 aws-java-sdk-core-protocols-aws- query-protocol 2.17.52

1.203.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by
Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.204 checker-qual 2.5.5

1.204.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

- * Copyright (c) 2005 Brian Goetz and Tim Peierls
- * Released under the Creative Commons Attribution License
- * (<http://creativecommons.org/licenses/by/2.5>)
- * Official home: <http://www.jcip.net>
- *
- * Any republication or derived work distributed in source code form
- * must include this copyright and license notice.
- */

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1046386584_1590042070.56/0/checker-qual-2-5-5-sources-jar/org/checkerframework/checker/lock/qual/GuardedBy.java

1.205 opencensus-api 0.28.0

1.205.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2018, OpenCensus Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/**
 * API for resource information population.
 *
 * <p>The resource library primarily defines a type "Resource" that captures information about the
 * entity for which stats or traces are recorded. For example, metrics exposed by a Kubernetes
 * container can be linked to a resource that specifies the cluster, namespace, pod, and container
 * name.
 *
 * <p>Two environment variables are used to populate resource information:
 *
 * <ul>
 * <li>OC_RESOURCE_TYPE: A string that describes the type of the resource prefixed by a domain
 * namespace. Leading and trailing whitespaces are trimmed. e.g. "kubernetes.io/container".
 * <li>OC_RESOURCE_LABELS: A comma-separated list of labels describing the source in more detail,
 * e.g. "key1=val1,key2=val2". The allowed character set is appropriately constrained.
 * </ul>
 *
 * <p>Type, label keys, and label values MUST contain only printable ASCII (codes between 32 and
 * 126, inclusive) and less than 256 characters. Type and label keys MUST have a length greater than
 * zero. They SHOULD start with a domain name and separate hierarchies with / characters, e.g.
 * k8s.io/namespace/name.
 *
 * <p>WARNING: Currently all the public classes under this package are marked as {@link
 * io.opencensus.common.ExperimentalApi}. DO NOT USE except for experimental purposes.
 *
 * <p>Please see
 * https://github.com/census-instrumentation/opencensus-specs/blob/master/resource/Resource.md for
```

* more details.

*/

Found in path(s):

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/resource/package-info.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2018, OpenCensus Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/metrics/DerivedLongGauge.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/BigendianEncoding.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/metrics/LabelKey.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/internal/Utils.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/ServerStatsDeserializationException.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/metrics/export/MetricProducer.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/package-info.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/metrics/export/Value.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/Tracestate.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/metrics/export/MetricProducerManager.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/metrics/export/Point.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/common/ServerStats.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/common/ToLongFunction.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/DoubleGauge.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/MetricsComponent.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/trace/package-info.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/LongGauge.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/DerivedDoubleGauge.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/export/Distribution.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/internal/DefaultVisibilityForTesting.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/Metrics.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/export/TimeSeries.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/export/ExportComponent.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/common/ToDoubleFunction.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/stats/package-info.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/resource/Resource.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/trace/MessageEvent.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/export/Summary.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/trace/BaseMessageEvent.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/LabelValue.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/export/Metric.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/common/ServerStatsEncoding.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/trace/internal/BaseMessageEventUtils.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/common/ServerStatsFieldEnums.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/export/MetricDescriptor.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/metrics/MetricRegistry.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2019, OpenCensus Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/tags/propagation/TagContextTextFormat.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/DerivedDoubleCumulative.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/DoubleCumulative.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/LongCumulative.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/data/AttachmentValue.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/DerivedLongCumulative.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/data/Exemplar.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/tags/TagMetadata.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/MetricOptions.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-
jar/io/opencensus/metrics/data/package-info.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2016-17, OpenCensus Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
```

*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/Tracing.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/ViewData.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/internal/StringUtils.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/View.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/AttributeValue.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/Measurement.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/Tracer.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/NetworkEvent.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/ViewManager.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/Span.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/Stats.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/Status.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/Timestamp.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/OpenCensusLibraryInformation.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/TraceId.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/internal/Provider.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/Function.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/BlankSpan.java
* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/trace/SpanContext.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/trace/TraceComponent.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/common/Duration.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017, OpenCensus Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/trace/SpanBuilder.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/tags/propagation/TagContextDeserializationException.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/trace/unsafe/ContextUtils.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/tags/TagsComponent.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/tags/TagValue.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/trace/config/TraceConfig.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/trace/propagation/SpanContextParseException.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/trace/Sampler.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/common/ExperimentalApi.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/trace/TraceOptions.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/trace/export/SpanData.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-

jar/io/opencensus/trace/samplers/Samplers.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/Functions.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/Measure.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/Tags.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/TagKey.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/internal/ZeroTimeClock.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/BucketBoundaries.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/EndSpanOptions.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/export/ExportComponent.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/AggregationData.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/Tag.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/export/SampledSpanStore.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/Internal.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/internal/NoopScope.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/InternalUtils.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/export/SpanExporter.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/propagation/TextFormat.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/StatsCollectionState.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/samplers/NeverSampleSampler.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/propagation/TagPropagationComponent.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/StatsRecorder.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/unsafe/ContextUtils.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/NoopTags.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/StatsComponent.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/TagContext.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/MeasureMap.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/Tagger.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/NoopStats.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/export/RunningSpanStore.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/SpanId.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/propagation/BinaryFormat.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/stats/Aggregation.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/package-info.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/propagation/TagContextSerializationException.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/Annotation.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/TaggingState.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/Link.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/CurrentSpanUtils.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/propagation/PropagationComponent.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/Clock.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/samplers/AlwaysSampleSampler.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/config/TraceParams.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/TagContextBuilder.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/internal/package-info.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/tags/propagation/TagContextBinarySerializer.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/NonThrowingCloseable.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/Scope.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/trace/samplers/ProbabilitySampler.java

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/common/TimeUtils.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2018, OpenCensus Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/**
 * This package describes the Metrics data model. Metrics are a data model for what stats exporters
 * take as input. This data model may eventually become the wire format for metrics.
 *
 * <p>WARNING: Currently all the public classes under this package are marked as { @link
 * io.opencensus.common.ExperimentalApi}. The classes and APIs under { @link io.opencensus.metrics }
 * are likely to get backwards-incompatible updates in the future. DO NOT USE except for
 * experimental purposes.
 *
 * <p>Please see
 * https://github.com/census-instrumentation/opencensus-specs/blob/master/stats/Metrics.md and
 * https://github.com/census-instrumentation/opencensus-
 * proto/blob/master/opencensus/proto/stats/metrics/metrics.proto
 * for more details.
 */
```

Found in path(s):

* /opt/cola/permits/1411867099_1662683857.9450252/0/opencensus-api-0-28-0-sources-1-jar/io/opencensus/metrics/package-info.java

1.206 io-grpc-grpc-stub 1.39.0

1.206.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent

to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,

excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any

risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2016-2020 Istio Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Envoy

Copyright 2016-2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object

form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a

file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner].

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

zero-allocation-hashing

Copyright 2015 Higher Frequency Trading <http://www.higherfrequencytrading.com>

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity

exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright 2015 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Copyright 2014 The gRPC Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This product contains a modified portion of 'OkHttp', an open source
HTTP & SPDY client for Android and Java applications, which can be obtained
at:

- * LICENSE:
 - * okhttp/third_party/okhttp/LICENSE (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/square/okhttp>
- * LOCATION_IN_GRPC:
 - * okhttp/third_party/okhttp

This product contains a modified portion of 'Envoy', an open source
cloud-native high-performance edge/middle/service proxy, which can be
obtained at:

- * LICENSE:
 - * xds/third_party/envoy/LICENSE (Apache License 2.0)
- * NOTICE:
 - * xds/third_party/envoy/NOTICE
- * HOMEPAGE:
 - * <https://www.envoyproxy.io>
- * LOCATION_IN_GRPC:
 - * xds/third_party/envoy

This product contains a modified portion of 'protoc-gen-validate (PGV)',
an open source protoc plugin to generate polyglot message validators,
which can be obtained at:

- * LICENSE:
 - * xds/third_party/protoc-gen-validate/LICENSE (Apache License 2.0)
- * NOTICE:

- * xds/third_party/protoc-gen-validate/NOTICE
- * HOMEPAGE:
 - * <https://github.com/envoyproxy/protoc-gen-validate>
- * LOCATION_IN_GRPC:
 - * xds/third_party/protoc-gen-validate

This product contains a modified portion of 'udpa',
an open source universal data plane API, which can be obtained at:

- * LICENSE:
 - * xds/third_party/udpa/LICENSE (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/cncf/udpa>
- * LOCATION_IN_GRPC:
 - * xds/third_party/udpa

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.207 netty/transport/classes/epoll

4.1.72.Final

1.207.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<!--  
~ Copyright 2014 The Netty Project  
~  
~ The Netty Project licenses this file to you under the Apache License,  
~ version 2.0 (the "License"); you may not use this file except in compliance  
~ with the License. You may obtain a copy of the License at:  
~  
~ https://www.apache.org/licenses/LICENSE-2.0  
~  
~ Unless required by applicable law or agreed to in writing, software  
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT  
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the  
~ License for the specific language governing permissions and limitations  
~ under the License.
```

-->

Found in path(s):

* /opt/cola/permits/1341130479_1654769335.3946347/0/netty-transport-native-epoll-4-1-72-final-jar/META-INF/maven/io.netty/netty-transport-native-epoll/pom.xml

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Implementation-Title: Netty/Transport/Native/Epoll

Bundle-Description: Netty is an asynchronous event-driven network application framework for rapid development of maintainable high performance protocol servers and clients.

Automatic-Module-Name: io.netty.transport.epoll

Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0>

Bundle-SymbolicName: io.netty.transport-native-epoll

Implementation-Version: 4.1.72.Final

Built-By: root

Bnd-LastModified: 1639393574219

Bundle-ManifestVersion: 2

Implementation-Vendor-Id: io.netty

Bundle-DocURL: <https://netty.io/>

Bundle-Vendor: The Netty Project

Import-Package: sun.nio.ch;resolution:=optional,org.eclipse.jetty.npn;version="[1,2)";resolution:=optional,org.eclipse.jetty.alpn;version="[1,2)";resolution:=optional

Tool: Bnd-2.4.1.201501161923

Implementation-Vendor: The Netty Project

Bundle-Name: Netty/Transport/Native/Epoll

Bundle-Version: 4.1.72.Final

Created-By: Apache Maven Bundle Plugin

Build-Jdk: 1.8.0_312

Implementation-URL: <https://netty.io/netty-transport-native-epoll/>

Found in path(s):

* /opt/cola/permits/1341130479_1654769335.3946347/0/netty-transport-native-epoll-4-1-72-final-jar/META-INF/MANIFEST.MF

1.208 apache-httpcomponents-core 4.4.13

1.208.1 Available under license :

Apache HttpComponents Core

Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<http://www.apache.org/>).

Apache License

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions

to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices

stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.209 aws-java-sdk-profiles 2.17.52

1.209.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed

with the accompanying software if such terms are included in the directory containing the accompanying software.

Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by
Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:

Apache Commons Lang

Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<https://www.apache.org/>).

1.210 json-java 20201115

1.210.1 Available under license :

Copyright (c) 2002 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.211 gopherjs 20181205-snapshot-0766667c

1.211.1 Available under license :

Copyright (c) 2016 Richard Musiol. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2013 Richard Musiol. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.212 mockito-junit-jupiter 3.11.2

1.212.1 Available under license :

The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.213 apache-avro 1.11.0

1.213.1 Available under license :

Apache Avro

Copyright 2009-2020 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.214 netty-project 4.1.72.Final

1.214.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Implementation-Title: Netty/Codec/HTTP

Bundle-Description: Netty is an asynchronous event-driven network appl
ication framework for rapid development of maintainable high perfo
rmance protocol servers and clients.

Automatic-Module-Name: io.netty.codec.http

Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0>

Bundle-SymbolicName: io.netty.codec-http

Implementation-Version: 4.1.72.Final

Built-By: norman

Bnd-LastModified: 1639395920992

Bundle-ManifestVersion: 2

Implementation-Vendor-Id: io.netty

Bundle-DocURL: <https://netty.io/>

Bundle-Vendor: The Netty Project

Import-Package: com.aayushatharva.brotli4j.encoder;resolution:=optiona
l,com.jcraft.jzlib;resolution:=optional,io.netty.buffer;version="[4.1
,5)",io.netty.channel;version="[4.1,5)",io.netty.channel.embedded;ver
sion="[4.1,5)",io.netty.handler.codec,io.netty.handler.codec.base64;v
ersion="[4.1,5)",io.netty.handler.codec.compression;version="[4.1,5)"
,io.netty.handler.ssl;version="[4.1,5)",io.netty.handler.stream;versi
on="[4.1,5)",io.netty.util;version="[4.1,5)",io.netty.util.concurrent
;version="[4.1,5)",io.netty.util.internal;version="[4.1,5)",io.netty.
util.internal.logging;version="[4.1,5)",sun.nio.ch;resolution:=option
al,org.eclipse.jetty.npn;version="[1,2)";resolution:=optional,org.ecl
ipse.jetty.alpn;version="[1,2)";resolution:=optional

Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.6))"

Tool: Bnd-2.4.1.201501161923

Implementation-Vendor: The Netty Project

Export-Package: io.netty.handler.codec.http;uses="io.netty.buffer,io.

io.netty.channel,io.netty.channel.embedded,io.netty.handler.codec,io.netty.handler.codec.compression,io.netty.handler.codec.http.cookie,io.netty.handler.stream,io.netty.util";version="4.1.72",io.netty.handler.codec.http.cookie;version="4.1.72",io.netty.handler.codec.http.cors;uses:="io.netty.channel,io.netty.handler.codec.http";version="4.1.72",io.netty.handler.codec.http.multipart;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec,io.netty.handler.codec.http,io.netty.handler.stream,io.netty.util";version="4.1.72",io.netty.handler.codec.http.websocketx;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec,io.netty.handler.codec.http,io.netty.handler.stream,io.netty.util,io.netty.util.internal.logging";version="4.1.72",io.netty.handler.codec.http.websocketx.extensions;uses:="io.netty.channel,io.netty.handler.codec,io.netty.handler.codec.http.websocketx";version="4.1.72",io.netty.handler.codec.http.websocketx.extensions.compression;uses:="io.netty.channel,io.netty.handler.codec.http.websocketx.extensions";version="4.1.72",io.netty.handler.codec.rtsp;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec.http,io.netty.util";version="4.1.72",io.netty.handler.codec.spdy;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec,io.netty.handler.codec.http,io.netty.util";version="4.1.72"

Bundle-Name: Netty/Codec/HTTP

Bundle-Version: 4.1.72.Final

Created-By: Apache Maven Bundle Plugin

Build-Jdk: 1.8.0_252

Implementation-URL: <https://netty.io/netty-codec-http/>

Found in path(s):

* /opt/cola/permits/1411867046_1662690214.5444846/0/netty-codec-http-4-1-72-final-jar/META-INF/MANIFEST.MF

No license file was found, but licenses were detected in source scan.

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):

* /opt/cola/permits/1411867046_1662690214.5444846/0/netty-codec-http-4-1-72-final-jar/META-INF/native-image/io.netty/codec-http/native-image.properties

No license file was found, but licenses were detected in source scan.

<!--

~ Copyright 2012 The Netty Project

~

~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:

~

~ <https://www.apache.org/licenses/LICENSE-2.0>
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->

Found in path(s):

* /opt/cola/permits/1411867046_1662690214.5444846/0/netty-codec-http-4-1-72-final-jar/META-INF/maven/io.netty/netty-codec-http/pom.xml

1.215 grpc-go v1.40.0

1.215.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.216 findbugs-jsr305 3.0.2

1.216.1 Available under license :

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

1. "Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be

considered a Derivative Work (as defined below) for the purposes of this License.

2. "Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

3. "Licensor" means the individual or entity that offers the Work under the terms of this License.

4. "Original Author" means the individual or entity who created the Work.

5. "Work" means the copyrightable work of authorship offered under the terms of this License.

6. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

1. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;

2. to create and reproduce Derivative Works;

3. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;

4. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

5.

For the avoidance of doubt, where the work is a musical composition:

1. Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

2. Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

6. Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

1. You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

2. If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

1. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

2. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

1. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

2. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

3. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

4. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

5. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

1.217 netty-transport-native-unix-common

4.1.68.Final

1.217.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License,

* version 2.0 (the "License"); you may not use this file except in compliance

* with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_jni.h
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/UnixChannelUtil.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2018 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_buffer.c
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/Buffer.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_buffer.h
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/PreferredDirectByteBufferAllocator.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance

* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/UnixChannelOption.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/Unix.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/package-info.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/IovArray.java

No license file was found, but licenses were detected in source scan.

<!--

~ Copyright 2016 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ <https://www.apache.org/licenses/LICENSE-2.0>
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->

Found in path(s):

* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/META-INF/maven/io.netty/netty-transport-native-unix-common/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2021 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance

* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/DomainDatagramChannel.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/DomainDatagramPacket.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/DomainDatagramSocketAddress.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/DomainDatagramChannelConfig.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/SegmentedDatagramPacket.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_limits.c
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_util.c
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/SocketWritableByteChannel.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-

1-jar/netty_unix_limits.h

* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/Limits.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/PeerCredentials.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_util.h
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/ErrorsStaticallyReferencedJniMethods.java
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/LimitsStaticallyReferencedJniMethods.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2020 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix.c
* /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix.h
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the

- * License for the specific language governing permissions and limitations
- * under the License.
- */

Found in path(s):

- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_filedescriptor.h
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/DomainSocketAddress.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/DomainSocketChannel.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/ServerDomainSocketChannel.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_errors.c
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_errors.h
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/FileDescriptor.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/DomainSocketChannelConfig.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/NativeInetAddress.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_socket.c
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/DatagramSocketAddress.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/Errors.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/UnixChannel.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_socket.h
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/Socket.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/io/netty/channel/unix/DomainSocketReadMode.java
- * /opt/cola/permits/1337069410_1654119310.1784396/0/netty-transport-native-unix-common-4-1-68-final-sources-1-jar/netty_unix_filedescriptor.c

1.218 everit-json-schema 1.12.2

1.218.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

Copyright (C) 2011 Everit Kft. (<http://www.everit.org>)

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->

Found in path(s):

* /opt/cola/permits/1411867003_1662683754.0293086/0/everit-json-schema-1-12-2-sources-2-jar/META-INF/maven/com.github.erosb/everit-json-schema/pom.xml

No license file was found, but licenses were detected in source scan.

/*

Copyright (c) 2002 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1411867003_1662683754.0293086/0/everit-json-schema-1-12-2-sources-2-jar/org/everit/json/schema/JSONPointer.java

* /opt/cola/permits/1411867003_1662683754.0293086/0/everit-json-schema-1-12-2-sources-2-jar/org/everit/json/schema/JSONPointerException.java

No license file was found, but licenses were detected in source scan.

/*

Copyright (c) 2006 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1411867003_1662683754.0293086/0/everit-json-schema-1-12-2-sources-2-jar/org/everit/json/schema/internal/JSONWriter.java

1.219 io-grpc-grpc-api 1.39.0

1.219.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2016-2020 Istio Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Envoy

Copyright 2016-2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner].

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

zero-allocation-hashing

Copyright 2015 Higher Frequency Trading <http://www.higherfrequencytrading.com>

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright 2015 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Copyright 2014 The gRPC Authors

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product contains a modified portion of 'OkHttp', an open source HTTP & SPDY client for Android and Java applications, which can be obtained at:

- * LICENSE:
 - * okhttp/third_party/okhttp/LICENSE (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/square/okhttp>
- * LOCATION_IN_GRPC:
 - * okhttp/third_party/okhttp

This product contains a modified portion of 'Envoy', an open source cloud-native high-performance edge/middle/service proxy, which can be obtained at:

- * LICENSE:
 - * xds/third_party/envoy/LICENSE (Apache License 2.0)
- * NOTICE:
 - * xds/third_party/envoy/NOTICE
- * HOMEPAGE:
 - * <https://www.envoyproxy.io>
- * LOCATION_IN_GRPC:
 - * xds/third_party/envoy

This product contains a modified portion of 'protoc-gen-validate (PGV)', an open source protoc plugin to generate polyglot message validators, which can be obtained at:

- * LICENSE:
 - * xds/third_party/protoc-gen-validate/LICENSE (Apache License 2.0)
- * NOTICE:
 - * xds/third_party/protoc-gen-validate/NOTICE
- * HOMEPAGE:
 - * <https://github.com/envoyproxy/protoc-gen-validate>
- * LOCATION_IN_GRPC:
 - * xds/third_party/protoc-gen-validate

This product contains a modified portion of 'udpa',
an open source universal data plane API, which can be obtained at:

- * LICENSE:
 - * xds/third_party/udpa/LICENSE (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/cncf/udpa>
- * LOCATION_IN_GRPC:
 - * xds/third_party/udpa

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work

(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses

granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"

replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.220 slf4j-api-module 1.7.30

1.220.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright (c) 2004-2007 QOS.ch

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2004-2013 QOS.ch

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or

otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents

of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).
Copyright (c) 2004-2017 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.221 io-grpc-grpc-protobuf-lite 1.39.0

1.221.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2014 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1183892729_1655096107.2169387/0/grpc-protobuf-lite-1-39-0-sources-
jar/io/grpc/protobuf/lite/ProtoLiteUtils.java
* /opt/cola/permits/1183892729_1655096107.2169387/0/grpc-protobuf-lite-1-39-0-sources-
jar/io/grpc/protobuf/lite/ProtoInputStream.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The gRPC Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1183892729_1655096107.2169387/0/grpc-protobuf-lite-1-39-0-sources-
jar/io/grpc/protobuf/lite/package-info.java

1.222 jackson-core 2.13.0

1.222.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.

It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0
To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.223 io-grpc-grpc-context 1.39.0

1.223.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,

publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution

notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing

the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2016-2020 Istio Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Envoy

Copyright 2016-2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to

communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of

the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner].

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

zero-allocation-hashing

Copyright 2015 Higher Frequency Trading <http://www.higherfrequencytrading.com>

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate

as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify

the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include

the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright 2015 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Copyright 2014 The gRPC Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product contains a modified portion of 'OkHttp', an open source HTTP & SPDY client for Android and Java applications, which can be obtained at:

- * LICENSE:
 - * okhttp/third_party/okhttp/LICENSE (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/square/okhttp>
- * LOCATION_IN_GRPC:
 - * okhttp/third_party/okhttp

This product contains a modified portion of 'Envoy', an open source cloud-native high-performance edge/middle/service proxy, which can be obtained at:

- * LICENSE:
 - * xds/third_party/envoy/LICENSE (Apache License 2.0)
- * NOTICE:
 - * xds/third_party/envoy/NOTICE
- * HOMEPAGE:
 - * <https://www.envoyproxy.io>
- * LOCATION_IN_GRPC:
 - * xds/third_party/envoy

This product contains a modified portion of 'protoc-gen-validate (PGV)', an open source protoc plugin to generate polyglot message validators, which can be obtained at:

- * LICENSE:
 - * xds/third_party/protoc-gen-validate/LICENSE (Apache License 2.0)
- * NOTICE:
 - * xds/third_party/protoc-gen-validate/NOTICE
- * HOMEPAGE:
 - * <https://github.com/envoyproxy/protoc-gen-validate>
- * LOCATION_IN_GRPC:
 - * xds/third_party/protoc-gen-validate

This product contains a modified portion of 'udpa', an open source universal data plane API, which can be obtained at:

- * LICENSE:
 - * xds/third_party/udpa/LICENSE (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/cncf/udpa>
- * LOCATION_IN_GRPC:
 - * xds/third_party/udpa

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.224 jmespath-go-jmespath v0.4.0

1.224.1 Available under license :

Copyright 2015 James Saryerwinnie

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.225 jacoco 0.8.7

1.225.1 Available under license :

Found license 'Eclipse Public License 1.0' in 'This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at'
Found license 'Eclipse Public License 1.0' in 'under the terms and conditions of the Eclipse Public License Version 2.0'

1.226 godoc-text v0.1.0

1.226.1 Available under license :

Copyright 2012 Keith Rarick

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.227 aws-java-sdk-for-aws-kms 1.12.26

1.227.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.228 aws-java-sdk-for-amazon-s3 1.12.26

1.228.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.229 kr-pretty 0.1.0

1.229.1 Available under license :

The MIT License (MIT)

Copyright 2012 Keith Rarick

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.230 asm-commons 9.1

1.230.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Bundle-DocURL: <http://asm.ow2.org>

Bundle-License: BSD-3-Clause;link=<https://asm.ow2.io/LICENSE.txt>

Bundle-ManifestVersion: 2

Bundle-Name: org.objectweb.asm.commons

Bundle-RequiredExecutionEnvironment: J2SE-1.5

Bundle-SymbolicName: org.objectweb.asm.commons

Bundle-Version: 9.1.0

Export-Package: org.objectweb.asm.commons;version="9.1";uses:="org.objectweb.asm,org.objectweb.asm.signature,org.objectweb.asm.tree"

Implementation-Title: Usefull class adapters based on ASM, a very small and fast Java bytecode manipulation framework

Implementation-Version: 9.1

Import-Package: org.objectweb.asm;version="[9.1,10)",org.objectweb.asm.signature;version="[9.1,10)",org.objectweb.asm.tree;version="[9.1,10)"

Module-Requires: org.objectweb.asm;transitive=true,org.objectweb.asm.tree;transitive=true,org.objectweb.asm.tree.analysis;transitive=true

Found in path(s):

* /opt/cola/permits/1183889887_1627494130.56/0/asm-commons-9-1-jar/META-INF/MANIFEST.MF

1.231 aws-java-sdk-::-core-::-protocols-::-

aws-json-protocol 2.17.52

1.231.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate

as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify

the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include

the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at

1.232 text v0.3.7

1.232.1 Available under license :

```
# This source code was written by the Go contributors.  
# The master list of contributors is in the main Go distribution,  
# visible at http://tip.golang.org/CONTRIBUTORS.  
Copyright (c) 2009 The Go Authors. All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.233 handy-uri-templates handy-uri-templates-2.1.8

1.233.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>
```

Found in path(s):

* /opt/cola/permits/1168630323_1662743882.4324734/0/handy-uri-templates-2-1-8-jar/META-INF/maven/com.damnhandy/handy-uri-templates/pom.xml

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Automatic-Module-Name: com.damnhandy.uri.template

Bnd-LastModified: 1558834247596

Build-Jdk: 11.0.3

Built-By: ryan

Bundle-Description: Handy URI Templates is a RFC6570 compliant URI template processor. The library allows clients to utilize templated URIs and inject replacement variables to expand the template into a URI. The library sports a fluent API, ability to plugin custom object renderers, and supports all levels of URI templates.

Bundle-DocURL: <https://github.com/damnhandy/Handy-URI-Templates>

Bundle-License: <http://www.apache.org/licenses/LICENSE-2.0.txt>

Bundle-ManifestVersion: 2

Bundle-Name: handy-uri-templates

Bundle-SymbolicName: com.damnhandy.handy-uri-templates

Bundle-Vendor: Ryan J. McDonough

Bundle-Version: 2.1.8

Created-By: Apache Maven Bundle Plugin

Export-Package: com.damnhandy.uri.template;version="2.1.8",com.damnhandy.uri.template.jackson.datatype;uses:="com.damnhandy.uri.template,com.fasterxml.jackson.core,com.fasterxml.jackson.databind,com.fasterxml.jackson.databind.module";version="2.1.8"

Import-Package: com.fasterxml.jackson.core;version="[2.9,3)";resolution:=optional,com.fasterxml.jackson.databind;version="[2.9,3)";resolution:=optional,com.fasterxml.jackson.databind.module;version="[2.9,3)";resolution:=optional,org.joda.time;version="[2.10,3)",org.joda.time.format;version="[2.10,3)"

key: value

mode: development

Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.7))"

Tool: Bnd-4.2.0.201903051501

url: <https://github.com/damnhandy/Handy-URI-Templates>

Found in path(s):

* /opt/cola/permits/1168630323_1662743882.4324734/0/handy-uri-templates-2-1-8-jar/META-INF/MANIFEST.MF

1.234 netty-reactive-streams-http-support 2.0.5

1.234.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bundle-Description: Reactive streams implementation for Netty.
Automatic-Module-Name: com.typesafe.netty.http
Bundle-License: <http://www.apache.org/licenses/LICENSE-2.0.txt>
Bundle-SymbolicName: com.typesafe.netty.reactive-streams-http
Built-By: marcospereira
Bnd-LastModified: 1602622977569
Bundle-ManifestVersion: 2
Bundle-DocURL: <http://typesafe.com/>
Bundle-Vendor: Typesafe
Import-Package: com.typesafe.netty;version="[2.0,3)",io.netty.buffer;version="[4.1,5)",io.netty.channel;version="[4.1,5)",io.netty.handler.codec;version="[4.1,5)",io.netty.handler.codec.http;version="[4.1,5)",io.netty.handler.codec.http.websocketx;version="[4.1,5)",io.netty.util;version="[4.1,5)",io.netty.util.concurrent;version="[4.1,5)",org.reactivestreams;version="[1.0,2)"
Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.7))"
Tool: Bnd-3.5.0.201709291849
Export-Package: com.typesafe.netty.http;uses:="io.netty.channel,io.netty.handler.codec.http,io.netty.handler.codec.http.websocketx,org.reactivestreams";version="2.0.5"
Bundle-Name: Netty Reactive Streams HTTP support
Bundle-Version: 2.0.5
Created-By: Apache Maven Bundle Plugin
Build-Jdk: 1.8.0_181

Found in path(s):

* /opt/cola/permits/1128619532_1649176822.43/0/netty-reactive-streams-http-2-0-5-jar/META-INF/MANIFEST.MF

1.235 github.com/xiaomi/pegasus-go-client 20210427-snapshot-f3b6b08b

1.235.1 Available under license :

Apache Pegasus
Copyright 2022 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally

submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or

implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

src/rdsn/** - MIT License

The MIT License (MIT)

Copyright (c) 2015 Microsoft Corporation

-- Robust Distributed System Nucleus (rDSN) --

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

src/shell/linenoise/* - BSD-2-Clause License

Copyright (c) 2010-2014, Salvatore Sanfilippo <antirez at gmail dot com>
Copyright (c) 2010-2013, Pieter Noordhuis <pnoordhuis at gmail dot com>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

src/shell/sds/* - BSD-2-Clause License

Copyright (c) 2006-2015, Salvatore Sanfilippo <antirez at gmail dot com>
Copyright (c) 2015, Oran Agra
Copyright (c) 2015, Redis Labs, Inc
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Redis nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

src/shell/argh.h - BSD-3-Clause License

Copyright (c) 2016, Adi Shavit
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

src/rdsn/include/dsn/utility/smart_pointers.h - Apache 2.0 License
src/rdsn/include/dsn/utility/string_view.h
src/rdsn/include/dsn/utility/absl/base/internal/invoke.h
src/rdsn/include/dsn/utility/absl/utility/utility.h
src/rdsn/src/utls/memutil.h
src/rdsn/src/utls/string_view.cpp
src/rdsn/src/utls/test/memutil_test.cpp
src/rdsn/src/utls/test/smart_pointers_test.cpp
src/rdsn/src/utls/test/string_view_test.cpp

Copyright 2017 The Abseil Authors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

src/rdsn/src/utis/safe_strerror_posix.cpp - BSD-3-Clause License

Copyright (c) 2006-2009 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

- * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

src/rdsn/include/dsn/utility/TokenBucket.h - Apache License, Version 2.0

src/rdsn/src/utis/test/TokenBucketTest.cpp

src/rdsn/src/utis/test/TokenBucketTest.h

Copyright (c) Facebook, Inc. and its affiliates.

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

src/rdsn/include/dsn/utility/hpc_locks/autoreseteventcondvar.h - zlib License

src/rdsn/include/dsn/utility/hpc_locks/rwlock.h

src/rdsn/include/dsn/utility/hpc_locks/autoresetevent.h

src/rdsn/include/dsn/utility/hpc_locks/sema.h

src/rdsn/include/dsn/utility/hpc_locks/bitfield.h

src/rdsn/include/dsn/utility/hpc_locks/benaphore.h

Copyright (c) 2015 Jeff Preshing

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgement in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

src/rdsn/src/runtime/build_config.h - BSD-3-Clause License

src/rdsn/src/utls/test/autoref_ptr_test.cpp

Copyright (c) 2012 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

cmake_modules/FindRT.cmake - BSD-3-Clause license
cmake_modules/FindDL.cmake

Copyright (c) 2010-2011, Rob Jansen

To the extent that a federal employee is an author of a portion of this software or a derivative work thereof, no copyright is claimed by the United States Government, as represented by the Secretary of the Navy ("GOVERNMENT") under Title 17, U.S. Code. All Other Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of the copyright owners nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. GOVERNMENT ALLOWS FREE USE OF THIS SOFTWARE IN ITS "AS IS" CONDITION AND DISCLAIMS ANY LIABILITY OF ANY KIND FOR ANY DAMAGES WHATSOEVER RESULTING FROM THE USE OF THIS SOFTWARE.

scripts/run-clang-format.py - MIT License

MIT License

Copyright (c) 2017 Guillaume Papin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

src/rdsn/include/dsn/utility/safe_strerror_posix.h - BSD-3-Clause License

Copyright (c) 2011 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/
Copyright (c) 2010-2014, Salvatore Sanfilippo <antirez at gmail dot com>
Copyright (c) 2010-2013, Pieter Noordhuis <pnoordhuis at gmail dot com>

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Pegasus is an effort undergoing incubation at The Apache Software Foundation (ASF),
sponsored by the Apache Incubator. Incubation is required of all newly accepted projects
until a further review indicates that the infrastructure, communications, and decision
making process have stabilized in a manner consistent with other successful ASF projects.
While incubation status is not necessarily a reflection of the completeness or stability

of the code, it does indicate that the project has yet to be fully endorsed by the ASF.

Some of the incubating project's releases may not be fully compliant with ASF policy. For example, releases may have incomplete or un-reviewed licensing conditions. What follows is a list of known issues the project is currently aware of (note that this list, by definition, is likely to be incomplete):

- * Releases may have incomplete licensing conditions.
- * Most of the top contributors have signed an ICLA and we are working on updating the headers.

If you are planning to incorporate this work into your product/project, please be aware that you will need to conduct a thorough licensing review to determine the overall implications of including this work. For the current status of this project through the Apache Incubator visit: <https://incubator.apache.org/projects/pegasus.html>

Copyright (c) 2006-2014, Salvatore Sanfilippo <antirez at gmail dot com>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.236 re2j 1.5

1.236.1 Available under license :

Copyright (c) \${year} The Go Authors. All rights reserved.

Use of this source code is governed by a BSD-style license that can be found in the LICENSE file.

This is a work derived from Russ Cox's RE2 in Go, whose license <http://golang.org/LICENSE> is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.237 aws-java-sdk-::-services-::-amazon-kinesis 2.17.52

1.237.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and

limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (<https://www.apache.org/>).

1.238 aws-java-sdk-http-clients-netty-non-blocking-i-o 2.17.52

1.238.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

AWS SDK for Java 2.0

Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by Amazon Technologies, Inc (<http://www.amazon.com/>).

THIRD PARTY COMPONENTS

This software includes third party software subject to the following copyrights:

- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.
- Apache Commons Lang - <https://github.com/apache/commons-lang>
- Netty Reactive Streams - <https://github.com/playframework/netty-reactive-streams>
- Jackson-core - <https://github.com/FasterXML/jackson-core>
- Jackson-dataformat-cbor - <https://github.com/FasterXML/jackson-dataformats-binary>

The licenses for these third party components are included in LICENSE.txt

- For Apache Commons Lang see also this required NOTICE:
Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (<https://www.apache.org/>).

1.239 golang-appengine v1.6.7

1.239.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.240 netty/tomcatnative-[openssl---classes]

2.0.46.Final

1.240.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<!--
~ Copyright 2021 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->
```

Found in path(s):

```
* /opt/cola/permits/1290493686_1654765977.1947134/0/netty-tcnative-classes-2-0-46-final-jar/META-INF/maven/io.netty/netty-tcnative-classes/pom.xml
```

1.241 kqueue 4.1.72.Final

1.241.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
```

*/

Found in path(s):

* /opt/cola/permits/1411866775_1662683909.5230677/0/netty-transport-native-kqueue-4-1-72-final-sources-1-jar/netty_kqueue_bsdsocket.h

* /opt/cola/permits/1411866775_1662683909.5230677/0/netty-transport-native-kqueue-4-1-72-final-sources-1-jar/netty_kqueue_eventarray.h

* /opt/cola/permits/1411866775_1662683909.5230677/0/netty-transport-native-kqueue-4-1-72-final-sources-1-jar/netty_kqueue_native.c

* /opt/cola/permits/1411866775_1662683909.5230677/0/netty-transport-native-kqueue-4-1-72-final-sources-1-jar/netty_kqueue_bsdsocket.c

* /opt/cola/permits/1411866775_1662683909.5230677/0/netty-transport-native-kqueue-4-1-72-final-sources-1-jar/netty_kqueue_eventarray.c

No license file was found, but licenses were detected in source scan.

<!--

~ Copyright 2016 The Netty Project

~

~ The Netty Project licenses this file to you under the Apache License,

~ version 2.0 (the "License"); you may not use this file except in compliance

~ with the License. You may obtain a copy of the License at:

~

~ <https://www.apache.org/licenses/LICENSE-2.0>

~

~ Unless required by applicable law or agreed to in writing, software

~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the

~ License for the specific language governing permissions and limitations

~ under the License.

-->

Found in path(s):

* /opt/cola/permits/1411866775_1662683909.5230677/0/netty-transport-native-kqueue-4-1-72-final-sources-1-jar/META-INF/maven/io.netty/netty-transport-native-kqueue/pom.xml

1.242 aws-glue-schema-registry-avro-serializer-deserializer 1.1.5

1.242.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

/*

* Copyright 2020 Amazon.com, Inc. or its affiliates.

* Licensed under the Apache License, Version 2.0 (the

* "License"); you may not use this file except in compliance

* with the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
-->

Found in path(s):

*/opt/cola/permits/1411866722_1662685569.8838844/0/schema-registry-serde-1-1-5-jar/META-INF/maven/software.amazon.glue/schema-registry-serde/pom.xml

1.243 vmihailenco/msgpack v4.0.4

1.243.1 Available under license :

Copyright (c) 2013 The github.com/vmihailenco/msgpack Authors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.244 aws-glue-schema-registry-build-tools

1.1.5

1.244.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<!--  
/*  
* Copyright 2020 Amazon.com, Inc. or its affiliates.  
* Licensed under the Apache License, Version 2.0 (the  
* "License"); you may not use this file except in compliance  
* with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/  
-->
```

Found in path(s):

```
* /opt/cola/permits/1411866962_1662685597.4012601/0/schema-registry-build-tools-1-1-5-jar/META-INF/maven/software.amazon.glue/schema-registry-build-tools/pom.xml
```

No license file was found, but licenses were detected in source scan.

```
<!--  
/*  
* Copyright 2019 Amazon.com, Inc. or its affiliates.  
* Licensed under the Apache License, Version 2.0 (the  
* "License"); you may not use this file except in compliance  
* with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/  
-->
```

Found in path(s):

- * /opt/cola/permits/1411866962_1662685597.4012601/0/schema-registry-build-tools-1-1-5-jar/checkstyle.xml
- * /opt/cola/permits/1411866962_1662685597.4012601/0/schema-registry-build-tools-1-1-5-jar/suppressions.xml

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

©2022 Cisco Systems, Inc. All rights reserved.