

=== Teste de Normalidade (Shapiro-Wilk) ===
IOA Mean:
Estatística W = 0.6294, p-valor = 0.000000
- Rejeita-se a hipótese de normalidade (p < 0.05)
IOA Without PA Mean:
Estatística W = 0.4876, p-valor = 0.000000
- Rejeita-se a hipótese de normalidade (p < 0.05)

=== Teste de Wilcoxon ===
Estatística W: 2326.0
p-valor: 0.603027
- Não há evidência de diferença estatística significativa (p ≥ 0.05)

=== Teste t de Student pareado ===
Quantidade de cenários comparados: 99
t-statistic: -3.9219
p-value: 0.000163
- Diferença estatisticamente significativa (p < 0.05)

=== Cenários com desvio padrão > 10% da média ===
IOA: 24 cenários com desvio > 10% da média
IOA-WITHOUT-PA: 18 cenários com desvio > 10% da média

- Cenários IOA com desvio alto:
- MPAndroidChart::com.github.mikephil.charting.rendrer.LineChartRenderer::drawData(Canvas) – 73.2% (63.2% acima de 10% da média)
- RxJava::rx.concurrency.TestScheduler::triggerActions(long) – 12.7% (2.7% acima de 10% da média)
- SimianArmy::com.netflix.simianarmy.basic.BasicSimianArmyContext::BasicSimianArmyContext(String) – 15.6% (5.6% acima de 10% da média)
- activiti::org.activiti.engine.impl.persistence.entity.DeploymentEntityManager::deleteDeployment(String, boolean) – 11.5% (1.5% acima de 10% da média)
- cloud-slang::io.cloudslang.lang.api.SlangImpl::getAllEventTypes() – 18.3% (8.3% acima de 10% da média)
- druid::com.metamx.druid.loading.S3SegmentPusher::push(File, DataSegment) – 14.1% (4.1% acima de 10% da média)
- elasticsearch::org.elasticsearch.action.DocWriteResponse::toXContent(XContentBuilder, Params) – 11.5% (1.5% acima de 10% da média)
- elasticsearch::org.elasticsearch.action.bulk.TransportShardBulkAction::update(IndexMetaData, BulkShardRequest, IndexShard, long[], VersionType[], Translog.Location, int, BulkItemRequest) – 13.3% (3.3% acima de 10% da média)
- elasticsearch::org.elasticsearch.action.index.TransportIndexAction::executeIndexRequestOnPrimary(IndexRequest, IndexShard, MappingUpdatedAction) – 10.3% (0.3% acima de 10% da média)
- elasticsearch::org.elasticsearch.action.support.replication.ReplicationOperation::execute() – 11.6% (1.6% acima de 10% da média)
- elasticsearch::org.elasticsearch.index.analysis.KurumojiAnalysisTests::createAnalysisService() – 18.4% (8.4% acima de 10% da média)
- elasticsearch::org.elasticsearch.index.analysis.commongrams.CommonGramsTokenFilterFactoryTests::testCommonGramsAnalysis() – 28.5% (18.5% acima de 10% da média)
- elasticsearch::org.elasticsearch.index.analysis.commongrams.CommonGramsTokenFilterFactoryTests::testQueryModeCommonGramsAnalysis() – 25.0% (15.0% acima de 10% da média)
- elasticsearch::org.elasticsearch.index.analysis.commongrams.CommonGramsTokenFilterFactoryTests::testSynonymsAnalysis() – 28.5% (18.5% acima de 10% da média)
- elasticsearch::org.elasticsearch.index.query.support.NestedInnerQueryParseSupport::setPathLevel() – 14.5% (4.5% acima de 10% da média)
- elasticsearch::org.elasticsearch.rest.action.cat.RestNodesAction::getTableWithHeader(RestRequest) – 11.1% (1.1% acima de 10% da média)
- elasticsearch::org.elasticsearch.test.ESSingleNodeTestCase::newNode() – 12.4% (2.4% acima de 10% da média)
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.MongoDBRiverDefinition::parseSettings(String, String, RiverSettings, ScriptService) – 12.8% (2.8% acima de 10% da média)
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.MongoDBRiverDefinition::parseSettings(String, String, RiverSettings, ScriptService) – 14.3% (4.3% acima de 10% da média)
- fitnesses::fitnesses.testsystems.slim.tables.SlimTableFactory::SlimTableFactory() – 13.5% (3.5% acima de 10% da média)
- jsoup::org.jsoup.nodes.TextNode::outerHtmlHead(StringBuilder, int, Document, OutputSettings) – 13.7% (3.7% acima de 10% da média)
- okhttp::com.squareup.okhttp.OkHttpClient::copyWithDefaults() – 11.9% (1.9% acima de 10% da média)
- spring-boot::org.springframework.boot.loader.archive.ExplodedArchive::getUrl() – 33.9% (23.9% acima de 10% da média)
- spring-boot::org.springframework.boot.loader.archive.ExplodedArchiveTests::getUrl() – 21.2% (11.2% acima de 10% da média)

- Cenários IOA-WITHOUT-PA com desvio alto:
- HikariCP::com.zaxxer.hikari.HikariConfig::validate() – 12.4% (2.4% acima de 10% da média)
- MPAndroidChart::com.github.mikephil.charting.renderer.LineChartRenderer::drawData(Canvas) – 35.1% (25.1% acima de 10% da média)
- MPAndroidChart::com.xmssasdeveloper.mpchartexample.ScatterChartActivity::onCreate(Bundle) – 13.0% (3.0% acima de 10% da média)
- MPAndroidChart::com.xmssasdeveloper.mpchartexample.ScatterChartActivity::onProgressChanged(SeekBar, int, boolean) – 11.9% (1.9% acima de 10% da média)
- SimianArmy::com.netflix.simianarmy.basic.BasicSimianArmyContext::BasicSimianArmyContext(String) – 13.3% (3.3% acima de 10% da média)
- SimianArmy::com.netflix.simianarmy.basic.janitor.BasicJanitorMonkeyContext::getInstanceJanitor() – 12.8% (2.8% acima de 10% da média)
- elasticsearch::org.elasticsearch.index.analysis.commongrams.CommonGramsTokenFilterFactoryTests::testCommonGramsAnalysis() – 21.1% (11.1% acima de 10% da média)
- elasticsearch::org.elasticsearch.index.analysis.commongrams.CommonGramsTokenFilterFactoryTests::testQueryModeCommonGramsAnalysis() – 19.9% (9.9% acima de 10% da média)
- elasticsearch::org.elasticsearch.index.analysis.commongrams.CommonGramsTokenFilterFactoryTests::testSynonymsAnalysis() – 11.1% (1.1% acima de 10% da média)
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.MongoDBRiverDefinition::parseSettings(String, String, RiverSettings, ScriptService) – 13.6% (3.6% acima de 10% da média)
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.Slurper::assignCollections() – 13.5% (3.5% acima de 10% da média)
- libgdx::com.badlogic.gdx.backends.lwjgl3.Lwjgl3Application::newWindow(ApplicationListener, Lwjgl3WindowConfiguration) – 13.3% (3.3% acima de 10% da média)
- libgdx::com.badlogic.gdx.backends.lwjgl3.Lwjgl3ApplicationConfiguration::copy(Lwjgl3ApplicationConfiguration) – 11.4% (1.4% acima de 10% da média)
- spring-boot::org.springframework.boot.loader.archive.ExplodedArchive::getUrl() – 37.2% (27.2% acima de 10% da média)
- spring-boot::org.springframework.boot.loader.archive.ExplodedArchiveTests::getUrl() – 166.5% (156.5% acima de 10% da média)
- storm::org.apache.storm.kafka.spout.KafkaSpout::open(Map, TopologyContext, SpoutOutputCollector) – 12.1% (2.1% acima de 10% da média)
- titan::com.thinkaurelius.titan.graphdb.transaction.StandardTitanTx::StandardTitanTx(StandardTitanGraph, TransactionConfiguration, BackendTransaction) – 11.2% (1.2% acima de 10% da média)
- voldemort::voldemort.server.VoldemortConfig::VoldemortConfig(Props) – 14.1% (4.1% acima de 10% da média)

- IOA
Maior desvio padrão: 1.85 (4.4% da média)
- Cenário: elasticsearch::org.elasticsearch.index.engine.InternalEngineTests::testRecoverFromForeignTranslog()
- Média correspondente: 42.13
Menor desvio padrão: 0.00 (25.0% da média)
- Cenário: elasticsearch::org.elasticsearch.index.analysis.commongrams.CommonGramsTokenFilterFactoryTests::testQueryModeCommonGramsAnalysis()
- Média correspondente: 0.00

- IOA Without PA
Maior desvio padrão: 6.96 (2.9% da média)
- Cenário: elasticsearch::org.elasticsearch.rest.action.cat.RestShardsAction::getTableWithHeader(RestRequest)
- Média correspondente: 242.18
Menor desvio padrão: 0.00 (8.1% da média)
- Cenário: okhttp::com.squareup.okhttp.OkHttpClient::copyWithDefaults()
- Média correspondente: 0.00

=== Média global de execução ===
IOA: 6.4691
IOA-WITHOUT-PA: 38.8051

=== Comparação cenário a cenário ===
Cenários em que IOA foi mais rápido: 34
Cenários em que IOA-WITHOUT-PA foi mais rápido: 65

- IOA mais rápido em:
- RxJava::rx.concurrency.TestScheduler::triggerActions(long) – 158.3260 segundos mais rápido
- RxJava::rx.internal.operators.OperatorMono::connect(Action1<?superSubscription>) – 291.7292 segundos mais rápido
- SimianArmy::com.netflix.simianarmy.basic.BasicSimianArmyContext::BasicSimianArmyContext(String) – 0.4427 segundos mais rápido
- activiti::org.activiti.engine.impl.bpmn.behavior.UserTaskActivityBehavior::execute(ActivityExecution) – 7.2079 segundos mais rápido
- activiti::org.activiti.engine.impl.persistence.entity.DeploymentEntityManager::deleteDeployment(String, boolean) – 8.4577 segundos mais rápido
- cloud-slang::io.cloudslang.lang.api.SlangImpl::getAllEventTypes() – 0.0430 segundos mais rápido
- elasticsearch::org.elasticsearch.action.DocWriteResponse::readFrom(StreamInput) – 64.5567 segundos mais rápido
- elasticsearch::org.elasticsearch.action.DocWriteResponse::toXContent(XContentBuilder, Params) – 62.6300 segundos mais rápido
- elasticsearch::org.elasticsearch.action.DocWriteResponse::writeTo(StreamOutput) – 65.7285 segundos mais rápido
- elasticsearch::org.elasticsearch.action.bulk.TransportShardBulkAction::update(IndexMetaData, BulkShardRequest, IndexShard, long[], VersionType[], Translog.Location, int, BulkItemRequest) – 74.9666 segundos mais rápido
- elasticsearch::org.elasticsearch.action.delete.TransportDeleteAction::executeDeleteRequestOnPrimary(DeleteRequest, IndexShard) – 69.7232 segundos mais rápido
- elasticsearch::org.elasticsearch.action.index.TransportIndexAction::executeIndexRequestOnPrimary(IndexRequest, IndexShard, MappingUpdatedAction) – 73.5648 segundos mais rápido
- elasticsearch::org.elasticsearch.action.support.replication.ReplicationOperation::execute() – 287.7790 segundos mais rápido
- elasticsearch::org.elasticsearch.action.support.replication.TransportReplicationAction::doRun() – 169.9226 segundos mais rápido
- elasticsearch::org.elasticsearch.action.update.TransportUpdateAction::shardOperation(UpdateRequest, ActionListener<UpdateResponse>, int) – 286.7265 segundos mais rápido
- elasticsearch::org.elasticsearch.cluster.metadata.MetaDataCreateIndexService::createIndex(CreateIndexClusterStateUpdateRequest, ActionListener<ClusterStateUpdateResponse>) – 265.1094 segundos mais rápido
- elasticsearch::org.elasticsearch.index.IndexSettings::IndexSettings(IndexMetaData, Settings, Predicate<String>, IndexScopedSettings) – 241.9116 segundos mais rápido
- elasticsearch::org.elasticsearch.index.analysis.KurumojiAnalysisTests::createAnalysisService() – 0.0009 segundos mais rápido
- elasticsearch::org.elasticsearch.index.analysis.commongrams.CommonGramsTokenFilterFactoryTests::testCommonGramsAnalysis() – 0.0004 segundos mais rápido
- elasticsearch::org.elasticsearch.index.analysis.commongrams.CommonGramsTokenFilterFactoryTests::testQueryModeCommonGramsAnalysis() – 0.0006 segundos mais rápido
- elasticsearch::org.elasticsearch.index.analysis.commongrams.CommonGramsTokenFilterFactoryTests::testSynonymsAnalysis() – 0.0002 segundos mais rápido
- elasticsearch::org.elasticsearch.index.engine.InternalEngine::InternalEngine(EngineConfig) – 67.5424 segundos mais rápido
- elasticsearch::org.elasticsearch.rest.action.cat.RestIndicesAction::getTableWithHeader(RestRequest) – 286.9432 segundos mais rápido
- elasticsearch::org.elasticsearch.rest.action.cat.RestNodesAction::getTableWithHeader(RestRequest) – 269.4150 segundos mais rápido
- elasticsearch::org.elasticsearch.rest.action.cat.RestShardsAction::getTableWithHeader(RestRequest) – 229.5189 segundos mais rápido
- hector::me.prettyprint.cassandra.connection.HConnectionManager::operateWithFailover(Operation<?>) – 2.4819 segundos mais rápido
- jenkins::jenkins.slaves.DefaultJnlpslaveReceiver::handle(String, JnlpslaveHandshake) – 8.7388 segundos mais rápido
- netty::org.jboss.netty.handler.codec.frame.LengthFieldBasedFrameDecoder::decode(ChannelHandlerContext, Channel, ChannelBuffer) – 4.9128 segundos mais rápido
- okhttp::com.squareup.okhttp.internal.bytes.GzipSource::consumeHeader(Deadline) – 2.6587 segundos mais rápido
- orientdb::com.orient technologies.orient.core.Orient::startup() – 282.0511 segundos mais rápido
- spring-boot::org.springframework.boot.loader.archive.ExplodedArchiveTests::getUrl() – 0.0028 segundos mais rápido
- titan::com.thinkaurelius.titan.graphdb.database.serialize.SerializerInitialization::initialize(Serializer) – 6.7055 segundos mais rápido
- voldemort::voldemort.VoldemortClientShell::VoldemortClientShell(ClientConfig, String, BufferedReader, PrintStream, PrintStream) – 81.8308 segundos mais rápido
- webbit::org.webbitserver.netty.WebSocketClient::adjustPipelineToWebsocket(ChannelHandlerContext, MessageEvent, ChannelHandler, ChannelHandler) – 3.9503 segundos mais rápido

- IOA-WITHOUT-PA mais rápido em:
- HikariCP::com.zaxxer.hikari.HikariConfig::validate() – 1.1923 segundos mais rápido
- HikariCP::com.zaxxer.hikari.HikariPool::addConnection() – 0.0410 segundos mais rápido
- MPAndroidChart::com.github.mikephil.charting.renderer.LineChartRenderer::drawData(Canvas) – 0.0006 segundos mais rápido
- MPAndroidChart::com.github.mikephil.charting.renderer.PieChartRenderer::drawData(Canvas) – 3.1562 segundos mais rápido
- MPAndroidChart::com.xmssasdeveloper.mpchartexample.LineChartActivity1::onCreate(Bundle) – 2.0064 segundos mais rápido
- MPAndroidChart::com.xmssasdeveloper.mpchartexample.LineChartActivity2::onCreate(Bundle) – 2.0020 segundos mais rápido
- MPAndroidChart::com.xmssasdeveloper.mpchartexample.RadarChartActivity::onCreate(Bundle) – 2.0079 segundos mais rápido
- MPAndroidChart::com.xmssasdeveloper.mpchartexample.ScatterChartActivity::onCreate(Bundle) – 2.3390 segundos mais rápido
- MPAndroidChart::com.xmssasdeveloper.mpchartexample.ScatterChartActivity::onProgressChanged(SeekBar, int, boolean) – 2.3243 segundos mais rápido
- MPAndroidChart::com.xmssasdeveloper.mpchartexample.StackedBarActivityNegative::onCreate(Bundle) – 2.0849 segundos mais rápido
- MPAndroidChart::com.xmssasdeveloper.mpchartexample.fragments.SimpleFragment::generateScatterData(int, float, int) – 2.4364 segundos mais rápido
- OpenTripPlanner::org.opentripplanner.routing.spt.MultiShortestPathTree::dominates(State, State) – 3.2270 segundos mais rápido
- SimianArmy::com.netflix.simianarmy.basic.janitor.BasicJanitorMonkeyContext::getInstanceJanitor() – 0.0815 segundos mais rápido
- alluxio::tachyon.hadoop.HdfsFileInputStream::HdfsFileInputStream(TachyonFS, int, Path, Configuration, int) – 0.8898 segundos mais rápido
- cucumber-jvm::cuke4duke.internal.java.JavaLanguage::load(String) – 0.3305 segundos mais rápido
- dropwizard::io.dropwizard.setup.Bootstrap::Bootstrap(Application<T>) – 0.5636 segundos mais rápido
- druid::com.metamx.druid.loading.S3SegmentPusher::push(File, DataSegment) – 0.3116 segundos mais rápido
- elasticsearch::org.elasticsearch.action.support.replication.ReplicationOperationTests::testReplication() – 38.9561 segundos mais rápido
- elasticsearch::org.elasticsearch.index.engine.InternalEngineTests::testRecoverFromForeignTranslog() – 37.1652 segundos mais rápido
- elasticsearch::org.elasticsearch.index.query.MultiMatchQueryBuilder::doXContent(XContentBuilder, Params) – 5.1422 segundos mais rápido
- elasticsearch::org.elasticsearch.index.query.SimpleIndexQueryParserTests::setup() – 2.6814 segundos mais rápido
- elasticsearch::org.elasticsearch.index.query.support.NestedInnerQueryParseSupport::setPathLevel() – 4.6775 segundos mais rápido
- elasticsearch::org.elasticsearch.test.ESSingleNodeTestCase::newNode() – 2.4601 segundos mais rápido
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.Indexer::applyAdvancedTransformation(QueueEntry, String) – 0.9839 segundos mais rápido
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.Indexer::processBlockingQueue(QueueEntry) – 0.8579 segundos mais rápido
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.MongoDBRiverDefinition::parseSettings(String, String, RiverSettings, ScriptService) – 1.1727 segundos mais rápido
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.MongoDBRiverDefinition::parseSettings(String, String, RiverSettings, ScriptService) – 0.8346 segundos mais rápido
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.Slurper::addToStream(Operation, Timestamp<?, DBObject, String) – 1.2634 segundos mais rápido
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.Slurper::assignCollections() – 1.4841 segundos mais rápido
- elasticsearch-river-mongodb::org.elasticsearch.river.mongodb.Slurper::run() – 0.3560 segundos mais rápido
- fitnesses::fitnesses.testsystems.slim.tables.SlimTableFactory::SlimTableFactory() – 2.3259 segundos mais rápido
- fitnesses::fitnesses.testsystems.slim.tables.SlimTableFactoryTest::setUp() – 4.9504 segundos mais rápido
- hector::me.prettyprint.cassandra.connection.client.HKerberosThriftClient::open() – 1.0807 segundos mais rápido
- hector::me.prettyprint.cassandra.connection.client.HSaslThriftClient::open() – 1.1061 segundos mais rápido
- hector::me.prettyprint.cassandra.connection.client.HThriftClient::open() – 1.1028 segundos mais rápido
- java-faker::com.github.javafaker.Faker::Faker(Locale, Random) – 0.6054 segundos mais rápido
- jsoup::org.jsoup.helper.HttpConnection::createConnection(Connection, Request) – 1.8117 segundos mais rápido
- jsoup::org.jsoup.helper.HttpConnection::execute(Connection, Request, Response) – 1.3207 segundos mais rápido
- jsoup::org.jsoup.helper.HttpConnection::execute(Connection, Request, Response) – 1.2287 segundos mais rápido
- jsoup::org.jsoup.nodes.TextNode::outerHtmlHead(StringBuilder, int, Document, OutputSettings) – 1.5362 segundos mais rápido
- libgdx::com.badlogic.gdx.backends.lwjgl3.Lwjgl3Application::newWindow(ApplicationListener, Lwjgl3WindowConfiguration) – 1.7842 segundos mais rápido
- libgdx::com.badlogic.gdx.backends.lwjgl3.Lwjgl3ApplicationConfiguration::copy(Lwjgl3ApplicationConfiguration) – 2.1447 segundos mais rápido
- logback::ch.qos.logback.core.rolling.TimeBasedFileNamingAndTriggeringPolicyBase::start() – 3.0542 segundos mais rápido
- okhttp::com.squareup.okhttp.OkHttpClient::copyWithDefaults() – 0.0005 segundos mais rápido
- pushy::com.relayrides.pushy.apns.PushManager::handleConnectionClosure(ApnsConnection<T>) – 0.6912 segundos mais rápido
- pushy::com.relayrides.pushy.apns.PushManager::handleConnectionFailure(ApnsConnection<T>, Throwable) – 0.8045 segundos mais rápido
- quickml::quickdt.experiments.OutOfTimeCrossValidatorRunner::getRandomForestBuilder(int, int) – 0.5165 segundos mais rápido
- resty-gwt::org.fusesource.restygwt.rebind.DirectRestServiceInterfaceClassCreator::getMethodCallback(JMethod) – 2.7157 segundos mais rápido
- retrofit::retrofit.RequestBuilder::build() – 0.5906 segundos mais rápido
- retrofit::retrofit.RestAdapter::logAndReplaceRequest(Request) – 0.8147 segundos mais rápido
- retrofit::retrofit.RestAdapter::logAndReplaceResponse(String, Response, long) – 0.8061 segundos mais rápido
- spring-boot::org.springframework.boot.autoconfigure.logging.AutoConfigurationReportLoggingInitializer::logAutoConfigurationReport(boolean) – 2.6594 segundos mais rápido
- spring-boot::org.springframework.boot.autoconfigure.mongo.MongoProperties::builder(MongoClientOptions) – 3.3082 segundos mais rápido
- spring-boot::org.springframework.boot.context.embedded.tomcat.TomcatEmbeddedServletContainerFactory::customizeConnector(Connector) – 0.6458 segundos mais rápido
- spring-boot::org.springframework.boot.context.embedded undertow.UndertowEmbeddedServletContainerFactory::createDeploymentManager(ServletContextInitializer) – 0.2604 segundos mais rápido
- spring-boot::org.springframework.boot.context.web.SpringBootServletInitializer::createRootApplicationContext(ServletContext) – 0.5119 segundos mais rápido
- spring-boot::org.springframework.boot.context.web.SpringBootServletInitializer::createRootApplicationContext(ServletContext) – 0.5804 segundos mais rápido
- spring-boot::org.springframework.boot.loader.archive.ExplodedArchive::getUrl() – 0.0003 segundos mais rápido
- storm::org.apache.storm.kafka.spout.KafkaSpout::emitTupleIfNotEmitted(ConsumerRecord<K,V>) – 0.6794 segundos mais rápido
- storm::org.apache.storm.kafka.spout.KafkaSpout::open(Map, TopologyContext, SpoutOutputCollector) – 1.0151 segundos mais rápido
- storm::org.apache.storm.kafka.spout.KafkaSpoutConfig::KafkaSpoutConfig(Builder<K,V>) – 1.0651 segundos mais rápido
- storm::org.apache.storm.kafka.spout.KafkaSpoutConfig::toString() – 0.9537 segundos mais rápido
- swagger-core::io.swagger.jaxrs2.OperationParser::getSchemaFromAnnotation(io.swagger.oas.annotations.media.Schema) – 0.8233 segundos mais rápido
- titan::com.thinkaurelius.titan.graphdb.transaction.StandardTitanTx::StandardTitanTx(StandardTitanGraph, TransactionConfiguration, BackendTransaction) – 0.5008 segundos mais rápido
- voldemort::voldemort.server.VoldemortConfig::VoldemortConfig(Props) – 7.2885 segundos mais rápido