Міністерство освіти і науки України

ДНІПРОВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені Олеся Гончара

Факультет фізики, електроніки та комп’ютерних систем

Кафедра електронних обчислювальних машин

Звіт

з лабораторної роботи №1

Виконав студент групи КІ-21-2 Шейко Р. О.

Керівник: Скуратовський І. А.

м. Дніпро

2024

Завдання до лабораторної роботи:

1. Написати програму в java, яка виконує наступне:

2. Створити колекцію із 10 мільйонів випадкових цілих значень в діапазоні від 1 до 100

3. за допомогою Stream API (кожне завдання робити окремим методом для того, щоб використати його в бенчмарку п. 4):

      - знайти суму елементів в колекції

      - знайти середнє значення елементів колекції

      - знайти стандартне відхилення

      - помножити на 2 кожен елемент вихідної колекції

      - відфільтрувати елементи колекції таким чином, щоб залишились лише парні значення, які діляться на 3 націло

4. Виконати ті самі дії за допомогою parallel stream

5. Оцінити затрачений час на кожну задачу із пп. 2-3 за допомогою JMH (https://github.com/openjdk/jmh). Бенчмарк методи не робити void, повертати з них розраховані значення (або якимось іншим способом запобігти оптимізації невикористовуваних значень).

6. Порівняльну таблицю роботи послідовного Stream API і parallel stream навести у звіті

7. У звіт додати посилання на Git репозиторій з кодом.

Почнемо з коду:

package main.com;  
  
import org.openjdk.jmh.annotations.\*;  
import java.util.List;  
import java.util.Random;  
import java.util.stream.Collectors;  
import java.util.stream.IntStream;  
  
@State(Scope.*Benchmark*)  
public class StreamBenchmark {  
  
 private static final int *SIZE* = 10\_000\_000;  
 private List<Integer> numbers;  
  
 @Setup  
 public void setup() {  
 Random random = new Random();  
 numbers = IntStream.*range*(0, *SIZE*)  
 .map(i -> random.nextInt(100) + 1)  
 .boxed()  
 .collect(Collectors.*toList*());  
 }  
  
 *// 1. Сума елементів у колекції* @Benchmark  
 public int sumSequential() {  
 return numbers.stream().mapToInt(Integer::intValue).sum();  
 }  
  
 @Benchmark  
 public int sumParallel() {  
 return numbers.parallelStream().mapToInt(Integer::intValue).sum();  
 }  
  
 *// 2. Середнє значення елементів колекції* @Benchmark  
 public double averageSequential() {  
 return numbers.stream().mapToInt(Integer::intValue).average().orElse(0.0);  
 }  
  
 @Benchmark  
 public double averageParallel() {  
 return numbers.parallelStream().mapToInt(Integer::intValue).average().orElse(0.0);  
 }  
  
 *// 3. Стандартне відхилення* @Benchmark  
 public double standardDeviationSequential() {  
 double avg = averageSequential();  
 return Math.*sqrt*(numbers.stream()  
 .mapToDouble(i -> Math.*pow*(i - avg, 2))  
 .average()  
 .orElse(0.0));  
 }  
  
 @Benchmark  
 public double standardDeviationParallel() {  
 double avg = averageParallel();  
 return Math.*sqrt*(numbers.parallelStream()  
 .mapToDouble(i -> Math.*pow*(i - avg, 2))  
 .average()  
 .orElse(0.0));  
 }  
  
 *// 4. Множення кожного елемента на 2* @Benchmark  
 public List<Integer> multiplySequential() {  
 return numbers.stream()  
 .map(i -> i \* 2)  
 .collect(Collectors.*toList*());  
 }  
  
 @Benchmark  
 public List<Integer> multiplyParallel() {  
 return numbers.parallelStream()  
 .map(i -> i \* 2)  
 .collect(Collectors.*toList*());  
 }  
  
 *// 5. Фільтрація: залишити парні значення, які діляться на 3* @Benchmark  
 public List<Integer> filterSequential() {  
 return numbers.stream()  
 .filter(i -> i % 2 == 0 && i % 3 == 0)  
 .collect(Collectors.*toList*());  
 }  
  
 @Benchmark  
 public List<Integer> filterParallel() {  
 return numbers.parallelStream()  
 .filter(i -> i % 2 == 0 && i % 3 == 0)  
 .collect(Collectors.*toList*());  
 }  
}

Спробуємо сформувати табличку(зад. 6):

|  |  |  |  |
| --- | --- | --- | --- |
| Операція | Паралельний потік (ops/s) | Послідовний потік (ops/s) | Прискорення (разів) |
| Середнє значення (average) | 567,370 ± 27,651 | 90,822 ± 1,387 | ~6,25 |
| Фільтрація (filter) | 38,244 ± 0,725 | 12,328 ± 0,050 | ~3,10 |
| Множення (multiply) | 2,273 ± 0,097 | 4,094 ± 0,153 | ~0,55 (швидше послідовний) |
| Стандартне відхилення | 194,967 ± 1,913 | 20,942 ± 0,050 | ~9,31 |
| Сума (sum) | 635,590 ± 5,816 | 156,121 ± 1,262 | ~4,07 |

Аналіз:

1. *Середнє значення, Фільтрація, Стандартне відхилення, Сума* — значно швидші у паралельному режимі, зокрема стандартне відхилення прискорюється в середньому у 9,31 разів.

2. *Множення* — єдина операція, де послідовний потік працює швидше, ніж паралельний. Це може бути пов'язано з низькою ефективністю розпаралелювання цієї операції або низькою обчислювальною складністю.

Така різниця пояснюється тим, що паралельні потоки можуть ефективніше використовувати багатоядерні процесори, особливо для завдань з високою обчислювальною складністю.

Посилання на GitHub:

<https://github.com/arg3ntuum/ABP>

Звіт в консолі

C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe "-javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath C:\Users\User\Documents\GitHub\ABP\Lab1\target\classes;C:\Users\User\.m2\repository\org\openjdk\jmh\jmh-core\1.36\jmh-core-1.36.jar;C:\Users\User\.m2\repository\net\sf\jopt-simple\jopt-simple\5.0.4\jopt-simple-5.0.4.jar;C:\Users\User\.m2\repository\org\apache\commons\commons-math3\3.2\commons-math3-3.2.jar main.com.Start

# JMH version: 1.36

# VM version: JDK 21.0.1, OpenJDK 64-Bit Server VM, 21.0.1+12-29

# VM invoker: C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe

# VM options: -javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8

# Blackhole mode: compiler (auto-detected, use -Djmh.blackhole.autoDetect=false to disable)

# Warmup: 5 iterations, 10 s each

# Measurement: 5 iterations, 10 s each

# Timeout: 10 min per iteration

# Threads: 1 thread, will synchronize iterations

# Benchmark mode: Throughput, ops/time

# Benchmark: main.com.StreamBenchmark.averageParallel

# Run progress: 0,00% complete, ETA 01:23:20

# Fork: 1 of 5

# Warmup Iteration 1: 474,589 ops/s

# Warmup Iteration 2: 546,645 ops/s

# Warmup Iteration 3: 579,205 ops/s

# Warmup Iteration 4: 587,917 ops/s

# Warmup Iteration 5: 567,528 ops/s

Iteration 1: 558,834 ops/s

Iteration 2: 547,411 ops/s

Iteration 3: 539,221 ops/s

Iteration 4: 541,714 ops/s

Iteration 5: 511,388 ops/s

# Run progress: 2,00% complete, ETA 01:22:55

# Fork: 2 of 5

# Warmup Iteration 1: 509,780 ops/s

# Warmup Iteration 2: 480,873 ops/s

# Warmup Iteration 3: 532,159 ops/s

# Warmup Iteration 4: 489,621 ops/s

# Warmup Iteration 5: 505,994 ops/s

Iteration 1: 531,993 ops/s

Iteration 2: 526,386 ops/s

Iteration 3: 517,424 ops/s

Iteration 4: 503,148 ops/s

Iteration 5: 522,040 ops/s

# Run progress: 4,00% complete, ETA 01:21:06

# Fork: 3 of 5

# Warmup Iteration 1: 441,246 ops/s

# Warmup Iteration 2: 561,205 ops/s

# Warmup Iteration 3: 563,801 ops/s

# Warmup Iteration 4: 571,944 ops/s

# Warmup Iteration 5: 589,635 ops/s

Iteration 1: 590,281 ops/s

Iteration 2: 599,612 ops/s

Iteration 3: 600,215 ops/s

Iteration 4: 624,940 ops/s

Iteration 5: 622,656 ops/s

# Run progress: 6,00% complete, ETA 01:19:22

# Fork: 4 of 5

# Warmup Iteration 1: 568,687 ops/s

# Warmup Iteration 2: 569,326 ops/s

# Warmup Iteration 3: 568,974 ops/s

# Warmup Iteration 4: 564,801 ops/s

# Warmup Iteration 5: 581,971 ops/s

Iteration 1: 572,205 ops/s

Iteration 2: 576,710 ops/s

Iteration 3: 542,125 ops/s

Iteration 4: 580,437 ops/s

Iteration 5: 572,840 ops/s

# Run progress: 8,00% complete, ETA 01:17:39

# Fork: 5 of 5

# Warmup Iteration 1: 576,925 ops/s

# Warmup Iteration 2: 600,086 ops/s

# Warmup Iteration 3: 606,688 ops/s

# Warmup Iteration 4: 610,211 ops/s

# Warmup Iteration 5: 597,743 ops/s

Iteration 1: 580,636 ops/s

Iteration 2: 604,022 ops/s

Iteration 3: 598,661 ops/s

Iteration 4: 609,033 ops/s

Iteration 5: 610,314 ops/s

Result "main.com.StreamBenchmark.averageParallel":

567,370 ±(99.9%) 27,651 ops/s [Average]

(min, avg, max) = (503,148, 567,370, 624,940), stdev = 36,913

CI (99.9%): [539,719, 595,021] (assumes normal distribution)

# JMH version: 1.36

# VM version: JDK 21.0.1, OpenJDK 64-Bit Server VM, 21.0.1+12-29

# VM invoker: C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe

# VM options: -javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8

# Blackhole mode: compiler (auto-detected, use -Djmh.blackhole.autoDetect=false to disable)

# Warmup: 5 iterations, 10 s each

# Measurement: 5 iterations, 10 s each

# Timeout: 10 min per iteration

# Threads: 1 thread, will synchronize iterations

# Benchmark mode: Throughput, ops/time

# Benchmark: main.com.StreamBenchmark.averageSequential

# Run progress: 10,00% complete, ETA 01:15:57

# Fork: 1 of 5

# Warmup Iteration 1: 92,084 ops/s

# Warmup Iteration 2: 92,134 ops/s

# Warmup Iteration 3: 93,944 ops/s

# Warmup Iteration 4: 94,576 ops/s

# Warmup Iteration 5: 91,961 ops/s

Iteration 1: 93,303 ops/s

Iteration 2: 89,648 ops/s

Iteration 3: 89,192 ops/s

Iteration 4: 89,658 ops/s

Iteration 5: 89,485 ops/s

# Run progress: 12,00% complete, ETA 01:14:16

# Fork: 2 of 5

# Warmup Iteration 1: 95,210 ops/s

# Warmup Iteration 2: 94,407 ops/s

# Warmup Iteration 3: 94,438 ops/s

# Warmup Iteration 4: 92,438 ops/s

# Warmup Iteration 5: 94,383 ops/s

Iteration 1: 93,604 ops/s

Iteration 2: 93,761 ops/s

Iteration 3: 93,470 ops/s

Iteration 4: 93,400 ops/s

Iteration 5: 92,637 ops/s

# Run progress: 14,00% complete, ETA 01:12:34

# Fork: 3 of 5

# Warmup Iteration 1: 89,374 ops/s

# Warmup Iteration 2: 87,573 ops/s

# Warmup Iteration 3: 89,990 ops/s

# Warmup Iteration 4: 90,430 ops/s

# Warmup Iteration 5: 91,071 ops/s

Iteration 1: 91,767 ops/s

Iteration 2: 91,155 ops/s

Iteration 3: 90,908 ops/s

Iteration 4: 90,377 ops/s

Iteration 5: 89,600 ops/s

# Run progress: 16,00% complete, ETA 01:10:52

# Fork: 4 of 5

# Warmup Iteration 1: 91,775 ops/s

# Warmup Iteration 2: 90,861 ops/s

# Warmup Iteration 3: 90,221 ops/s

# Warmup Iteration 4: 90,387 ops/s

# Warmup Iteration 5: 91,032 ops/s

Iteration 1: 90,374 ops/s

Iteration 2: 87,500 ops/s

Iteration 3: 89,315 ops/s

Iteration 4: 88,405 ops/s

Iteration 5: 87,383 ops/s

# Run progress: 18,00% complete, ETA 01:09:10

# Fork: 5 of 5

# Warmup Iteration 1: 91,142 ops/s

# Warmup Iteration 2: 93,242 ops/s

# Warmup Iteration 3: 90,082 ops/s

# Warmup Iteration 4: 90,966 ops/s

# Warmup Iteration 5: 91,299 ops/s

Iteration 1: 92,088 ops/s

Iteration 2: 91,188 ops/s

Iteration 3: 90,903 ops/s

Iteration 4: 90,682 ops/s

Iteration 5: 90,737 ops/s

Result "main.com.StreamBenchmark.averageSequential":

90,822 ±(99.9%) 1,387 ops/s [Average]

(min, avg, max) = (87,383, 90,822, 93,761), stdev = 1,852

CI (99.9%): [89,434, 92,209] (assumes normal distribution)

# JMH version: 1.36

# VM version: JDK 21.0.1, OpenJDK 64-Bit Server VM, 21.0.1+12-29

# VM invoker: C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe

# VM options: -javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8

# Blackhole mode: compiler (auto-detected, use -Djmh.blackhole.autoDetect=false to disable)

# Warmup: 5 iterations, 10 s each

# Measurement: 5 iterations, 10 s each

# Timeout: 10 min per iteration

# Threads: 1 thread, will synchronize iterations

# Benchmark mode: Throughput, ops/time

# Benchmark: main.com.StreamBenchmark.filterParallel

# Run progress: 20,00% complete, ETA 01:07:29

# Fork: 1 of 5

# Warmup Iteration 1: 36,035 ops/s

# Warmup Iteration 2: 36,664 ops/s

# Warmup Iteration 3: 38,872 ops/s

# Warmup Iteration 4: 38,926 ops/s

# Warmup Iteration 5: 38,153 ops/s

Iteration 1: 39,137 ops/s

Iteration 2: 37,005 ops/s

Iteration 3: 37,818 ops/s

Iteration 4: 38,142 ops/s

Iteration 5: 38,714 ops/s

# Run progress: 22,00% complete, ETA 01:05:49

# Fork: 2 of 5

# Warmup Iteration 1: 33,911 ops/s

# Warmup Iteration 2: 33,933 ops/s

# Warmup Iteration 3: 38,472 ops/s

# Warmup Iteration 4: 38,692 ops/s

# Warmup Iteration 5: 36,800 ops/s

Iteration 1: 38,488 ops/s

Iteration 2: 39,221 ops/s

Iteration 3: 37,925 ops/s

Iteration 4: 37,944 ops/s

Iteration 5: 36,671 ops/s

# Run progress: 24,00% complete, ETA 01:04:08

# Fork: 3 of 5

# Warmup Iteration 1: 36,433 ops/s

# Warmup Iteration 2: 37,109 ops/s

# Warmup Iteration 3: 38,745 ops/s

# Warmup Iteration 4: 38,140 ops/s

# Warmup Iteration 5: 37,343 ops/s

Iteration 1: 37,449 ops/s

Iteration 2: 38,636 ops/s

Iteration 3: 38,441 ops/s

Iteration 4: 39,087 ops/s

Iteration 5: 38,758 ops/s

# Run progress: 26,00% complete, ETA 01:02:27

# Fork: 4 of 5

# Warmup Iteration 1: 36,848 ops/s

# Warmup Iteration 2: 38,135 ops/s

# Warmup Iteration 3: 38,150 ops/s

# Warmup Iteration 4: 37,756 ops/s

# Warmup Iteration 5: 38,748 ops/s

Iteration 1: 35,519 ops/s

Iteration 2: 37,866 ops/s

Iteration 3: 37,071 ops/s

Iteration 4: 38,449 ops/s

Iteration 5: 38,807 ops/s

# Run progress: 28,00% complete, ETA 01:00:46

# Fork: 5 of 5

# Warmup Iteration 1: 37,557 ops/s

# Warmup Iteration 2: 37,885 ops/s

# Warmup Iteration 3: 39,289 ops/s

# Warmup Iteration 4: 38,943 ops/s

# Warmup Iteration 5: 39,148 ops/s

Iteration 1: 39,491 ops/s

Iteration 2: 39,030 ops/s

Iteration 3: 39,514 ops/s

Iteration 4: 39,187 ops/s

Iteration 5: 37,742 ops/s

Result "main.com.StreamBenchmark.filterParallel":

38,244 ±(99.9%) 0,725 ops/s [Average]

(min, avg, max) = (35,519, 38,244, 39,514), stdev = 0,967

CI (99.9%): [37,520, 38,969] (assumes normal distribution)

# JMH version: 1.36

# VM version: JDK 21.0.1, OpenJDK 64-Bit Server VM, 21.0.1+12-29

# VM invoker: C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe

# VM options: -javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8

# Blackhole mode: compiler (auto-detected, use -Djmh.blackhole.autoDetect=false to disable)

# Warmup: 5 iterations, 10 s each

# Measurement: 5 iterations, 10 s each

# Timeout: 10 min per iteration

# Threads: 1 thread, will synchronize iterations

# Benchmark mode: Throughput, ops/time

# Benchmark: main.com.StreamBenchmark.filterSequential

# Run progress: 30,00% complete, ETA 00:59:05

# Fork: 1 of 5

# Warmup Iteration 1: 12,311 ops/s

# Warmup Iteration 2: 12,359 ops/s

# Warmup Iteration 3: 12,427 ops/s

# Warmup Iteration 4: 12,368 ops/s

# Warmup Iteration 5: 12,382 ops/s

Iteration 1: 12,276 ops/s

Iteration 2: 12,372 ops/s

Iteration 3: 12,424 ops/s

Iteration 4: 12,385 ops/s

Iteration 5: 12,366 ops/s

# Run progress: 32,00% complete, ETA 00:57:24

# Fork: 2 of 5

# Warmup Iteration 1: 12,325 ops/s

# Warmup Iteration 2: 12,330 ops/s

# Warmup Iteration 3: 12,447 ops/s

# Warmup Iteration 4: 12,400 ops/s

# Warmup Iteration 5: 12,477 ops/s

Iteration 1: 12,289 ops/s

Iteration 2: 12,414 ops/s

Iteration 3: 12,264 ops/s

Iteration 4: 12,374 ops/s

Iteration 5: 12,387 ops/s

# Run progress: 34,00% complete, ETA 00:55:43

# Fork: 3 of 5

# Warmup Iteration 1: 12,263 ops/s

# Warmup Iteration 2: 12,314 ops/s

# Warmup Iteration 3: 12,256 ops/s

# Warmup Iteration 4: 12,329 ops/s

# Warmup Iteration 5: 12,271 ops/s

Iteration 1: 12,302 ops/s

Iteration 2: 12,323 ops/s

Iteration 3: 12,301 ops/s

Iteration 4: 12,193 ops/s

Iteration 5: 12,210 ops/s

# Run progress: 36,00% complete, ETA 00:54:02

# Fork: 4 of 5

# Warmup Iteration 1: 12,318 ops/s

# Warmup Iteration 2: 12,379 ops/s

# Warmup Iteration 3: 12,408 ops/s

# Warmup Iteration 4: 12,386 ops/s

# Warmup Iteration 5: 12,297 ops/s

Iteration 1: 12,293 ops/s

Iteration 2: 12,282 ops/s

Iteration 3: 12,430 ops/s

Iteration 4: 12,397 ops/s

Iteration 5: 12,393 ops/s

# Run progress: 38,00% complete, ETA 00:52:22

# Fork: 5 of 5

# Warmup Iteration 1: 12,230 ops/s

# Warmup Iteration 2: 12,383 ops/s

# Warmup Iteration 3: 12,384 ops/s

# Warmup Iteration 4: 12,377 ops/s

# Warmup Iteration 5: 12,344 ops/s

Iteration 1: 12,354 ops/s

Iteration 2: 12,253 ops/s

Iteration 3: 12,251 ops/s

Iteration 4: 12,315 ops/s

Iteration 5: 12,356 ops/s

Result "main.com.StreamBenchmark.filterSequential":

12,328 ±(99.9%) 0,050 ops/s [Average]

(min, avg, max) = (12,193, 12,328, 12,430), stdev = 0,067

CI (99.9%): [12,278, 12,378] (assumes normal distribution)

# JMH version: 1.36

# VM version: JDK 21.0.1, OpenJDK 64-Bit Server VM, 21.0.1+12-29

# VM invoker: C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe

# VM options: -javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8

# Blackhole mode: compiler (auto-detected, use -Djmh.blackhole.autoDetect=false to disable)

# Warmup: 5 iterations, 10 s each

# Measurement: 5 iterations, 10 s each

# Timeout: 10 min per iteration

# Threads: 1 thread, will synchronize iterations

# Benchmark mode: Throughput, ops/time

# Benchmark: main.com.StreamBenchmark.multiplyParallel

# Run progress: 40,00% complete, ETA 00:50:40

# Fork: 1 of 5

# Warmup Iteration 1: 2,143 ops/s

# Warmup Iteration 2: 2,318 ops/s

# Warmup Iteration 3: 2,307 ops/s

# Warmup Iteration 4: 2,029 ops/s

# Warmup Iteration 5: 2,359 ops/s

Iteration 1: 2,257 ops/s

Iteration 2: 2,353 ops/s

Iteration 3: 2,182 ops/s

Iteration 4: 2,210 ops/s

Iteration 5: 2,405 ops/s

# Run progress: 42,00% complete, ETA 00:49:03

# Fork: 2 of 5

# Warmup Iteration 1: 2,131 ops/s

# Warmup Iteration 2: 2,059 ops/s

# Warmup Iteration 3: 2,358 ops/s

# Warmup Iteration 4: 2,344 ops/s

# Warmup Iteration 5: 2,238 ops/s

Iteration 1: 2,245 ops/s

Iteration 2: 2,282 ops/s

Iteration 3: 2,278 ops/s

Iteration 4: 2,124 ops/s

Iteration 5: 2,360 ops/s

# Run progress: 44,00% complete, ETA 00:47:26

# Fork: 3 of 5

# Warmup Iteration 1: 2,087 ops/s

# Warmup Iteration 2: 2,314 ops/s

# Warmup Iteration 3: 2,437 ops/s

# Warmup Iteration 4: 2,156 ops/s

# Warmup Iteration 5: 2,331 ops/s

Iteration 1: 1,990 ops/s

Iteration 2: 2,063 ops/s

Iteration 3: 2,358 ops/s

Iteration 4: 2,150 ops/s

Iteration 5: 2,293 ops/s

# Run progress: 46,00% complete, ETA 00:45:46

# Fork: 4 of 5

# Warmup Iteration 1: 2,243 ops/s

# Warmup Iteration 2: 2,385 ops/s

# Warmup Iteration 3: 2,449 ops/s

# Warmup Iteration 4: 2,646 ops/s

# Warmup Iteration 5: 2,543 ops/s

Iteration 1: 2,422 ops/s

Iteration 2: 2,433 ops/s

Iteration 3: 2,389 ops/s

Iteration 4: 2,488 ops/s

Iteration 5: 2,342 ops/s

# Run progress: 48,00% complete, ETA 00:44:06

# Fork: 5 of 5

# Warmup Iteration 1: 1,893 ops/s

# Warmup Iteration 2: 2,319 ops/s

# Warmup Iteration 3: 2,009 ops/s

# Warmup Iteration 4: 2,176 ops/s

# Warmup Iteration 5: 2,159 ops/s

Iteration 1: 2,147 ops/s

Iteration 2: 2,071 ops/s

Iteration 3: 2,264 ops/s

Iteration 4: 2,365 ops/s

Iteration 5: 2,364 ops/s

Result "main.com.StreamBenchmark.multiplyParallel":

2,273 ±(99.9%) 0,097 ops/s [Average]

(min, avg, max) = (1,990, 2,273, 2,488), stdev = 0,129

CI (99.9%): [2,177, 2,370] (assumes normal distribution)

# JMH version: 1.36

# VM version: JDK 21.0.1, OpenJDK 64-Bit Server VM, 21.0.1+12-29

# VM invoker: C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe

# VM options: -javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8

# Blackhole mode: compiler (auto-detected, use -Djmh.blackhole.autoDetect=false to disable)

# Warmup: 5 iterations, 10 s each

# Measurement: 5 iterations, 10 s each

# Timeout: 10 min per iteration

# Threads: 1 thread, will synchronize iterations

# Benchmark mode: Throughput, ops/time

# Benchmark: main.com.StreamBenchmark.multiplySequential

# Run progress: 50,00% complete, ETA 00:42:26

# Fork: 1 of 5

# Warmup Iteration 1: 4,258 ops/s

# Warmup Iteration 2: 3,981 ops/s

# Warmup Iteration 3: 4,395 ops/s

# Warmup Iteration 4: 4,150 ops/s

# Warmup Iteration 5: 4,096 ops/s

Iteration 1: 4,261 ops/s

Iteration 2: 4,102 ops/s

Iteration 3: 3,856 ops/s

Iteration 4: 3,657 ops/s

Iteration 5: 4,289 ops/s

# Run progress: 52,00% complete, ETA 00:40:45

# Fork: 2 of 5

# Warmup Iteration 1: 4,457 ops/s

# Warmup Iteration 2: 4,476 ops/s

# Warmup Iteration 3: 4,412 ops/s

# Warmup Iteration 4: 4,154 ops/s

# Warmup Iteration 5: 4,049 ops/s

Iteration 1: 4,258 ops/s

Iteration 2: 3,686 ops/s

Iteration 3: 4,068 ops/s

Iteration 4: 3,962 ops/s

Iteration 5: 4,322 ops/s

# Run progress: 54,00% complete, ETA 00:39:04

# Fork: 3 of 5

# Warmup Iteration 1: 4,011 ops/s

# Warmup Iteration 2: 4,188 ops/s

# Warmup Iteration 3: 4,192 ops/s

# Warmup Iteration 4: 4,026 ops/s

# Warmup Iteration 5: 4,550 ops/s

Iteration 1: 4,465 ops/s

Iteration 2: 4,155 ops/s

Iteration 3: 4,343 ops/s

Iteration 4: 4,066 ops/s

Iteration 5: 4,051 ops/s

# Run progress: 56,00% complete, ETA 00:37:23

# Fork: 4 of 5

# Warmup Iteration 1: 4,288 ops/s

# Warmup Iteration 2: 4,149 ops/s

# Warmup Iteration 3: 4,198 ops/s

# Warmup Iteration 4: 3,890 ops/s

# Warmup Iteration 5: 4,262 ops/s

Iteration 1: 4,060 ops/s

Iteration 2: 3,971 ops/s

Iteration 3: 4,088 ops/s

Iteration 4: 4,198 ops/s

Iteration 5: 4,268 ops/s

# Run progress: 58,00% complete, ETA 00:35:41

# Fork: 5 of 5

# Warmup Iteration 1: 4,689 ops/s

# Warmup Iteration 2: 4,667 ops/s

# Warmup Iteration 3: 4,271 ops/s

# Warmup Iteration 4: 4,271 ops/s

# Warmup Iteration 5: 4,197 ops/s

Iteration 1: 3,830 ops/s

Iteration 2: 4,096 ops/s

Iteration 3: 3,986 ops/s

Iteration 4: 3,969 ops/s

Iteration 5: 4,350 ops/s

Result "main.com.StreamBenchmark.multiplySequential":

4,094 ±(99.9%) 0,153 ops/s [Average]

(min, avg, max) = (3,657, 4,094, 4,465), stdev = 0,204

CI (99.9%): [3,941, 4,248] (assumes normal distribution)

# JMH version: 1.36

# VM version: JDK 21.0.1, OpenJDK 64-Bit Server VM, 21.0.1+12-29

# VM invoker: C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe

# VM options: -javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8

# Blackhole mode: compiler (auto-detected, use -Djmh.blackhole.autoDetect=false to disable)

# Warmup: 5 iterations, 10 s each

# Measurement: 5 iterations, 10 s each

# Timeout: 10 min per iteration

# Threads: 1 thread, will synchronize iterations

# Benchmark mode: Throughput, ops/time

# Benchmark: main.com.StreamBenchmark.standardDeviationParallel

# Run progress: 60,00% complete, ETA 00:34:00

# Fork: 1 of 5

# Warmup Iteration 1: 190,613 ops/s

# Warmup Iteration 2: 179,131 ops/s

# Warmup Iteration 3: 189,550 ops/s

# Warmup Iteration 4: 195,707 ops/s

# Warmup Iteration 5: 195,844 ops/s

Iteration 1: 193,536 ops/s

Iteration 2: 195,686 ops/s

Iteration 3: 188,695 ops/s

Iteration 4: 193,019 ops/s

Iteration 5: 195,831 ops/s

# Run progress: 62,00% complete, ETA 00:32:17

# Fork: 2 of 5

# Warmup Iteration 1: 191,203 ops/s

# Warmup Iteration 2: 193,487 ops/s

# Warmup Iteration 3: 195,149 ops/s

# Warmup Iteration 4: 188,902 ops/s

# Warmup Iteration 5: 195,401 ops/s

Iteration 1: 193,677 ops/s

Iteration 2: 196,607 ops/s

Iteration 3: 196,789 ops/s

Iteration 4: 194,100 ops/s

Iteration 5: 195,933 ops/s

# Run progress: 64,00% complete, ETA 00:30:35

# Fork: 3 of 5

# Warmup Iteration 1: 191,666 ops/s

# Warmup Iteration 2: 194,407 ops/s

# Warmup Iteration 3: 194,586 ops/s

# Warmup Iteration 4: 196,995 ops/s

# Warmup Iteration 5: 197,077 ops/s

Iteration 1: 188,203 ops/s

Iteration 2: 196,776 ops/s

Iteration 3: 197,015 ops/s

Iteration 4: 195,015 ops/s

Iteration 5: 197,141 ops/s

# Run progress: 66,00% complete, ETA 00:28:52

# Fork: 4 of 5

# Warmup Iteration 1: 191,536 ops/s

# Warmup Iteration 2: 186,032 ops/s

# Warmup Iteration 3: 194,917 ops/s

# Warmup Iteration 4: 197,033 ops/s

# Warmup Iteration 5: 196,178 ops/s

Iteration 1: 195,461 ops/s

Iteration 2: 196,736 ops/s

Iteration 3: 190,636 ops/s

Iteration 4: 195,299 ops/s

Iteration 5: 197,473 ops/s

# Run progress: 68,00% complete, ETA 00:27:10

# Fork: 5 of 5

# Warmup Iteration 1: 192,685 ops/s

# Warmup Iteration 2: 195,007 ops/s

# Warmup Iteration 3: 193,086 ops/s

# Warmup Iteration 4: 191,863 ops/s

# Warmup Iteration 5: 197,355 ops/s

Iteration 1: 194,069 ops/s

Iteration 2: 197,121 ops/s

Iteration 3: 197,566 ops/s

Iteration 4: 195,512 ops/s

Iteration 5: 196,276 ops/s

Result "main.com.StreamBenchmark.standardDeviationParallel":

194,967 ±(99.9%) 1,913 ops/s [Average]

(min, avg, max) = (188,203, 194,967, 197,566), stdev = 2,553

CI (99.9%): [193,054, 196,880] (assumes normal distribution)

# JMH version: 1.36

# VM version: JDK 21.0.1, OpenJDK 64-Bit Server VM, 21.0.1+12-29

# VM invoker: C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe

# VM options: -javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8

# Blackhole mode: compiler (auto-detected, use -Djmh.blackhole.autoDetect=false to disable)

# Warmup: 5 iterations, 10 s each

# Measurement: 5 iterations, 10 s each

# Timeout: 10 min per iteration

# Threads: 1 thread, will synchronize iterations

# Benchmark mode: Throughput, ops/time

# Benchmark: main.com.StreamBenchmark.standardDeviationSequential

# Run progress: 70,00% complete, ETA 00:25:28

# Fork: 1 of 5

# Warmup Iteration 1: 20,907 ops/s

# Warmup Iteration 2: 20,980 ops/s

# Warmup Iteration 3: 20,980 ops/s

# Warmup Iteration 4: 20,967 ops/s

# Warmup Iteration 5: 20,935 ops/s

Iteration 1: 20,905 ops/s

Iteration 2: 21,005 ops/s

Iteration 3: 20,898 ops/s

Iteration 4: 20,985 ops/s

Iteration 5: 20,980 ops/s

# Run progress: 72,00% complete, ETA 00:23:46

# Fork: 2 of 5

# Warmup Iteration 1: 20,845 ops/s

# Warmup Iteration 2: 20,811 ops/s

# Warmup Iteration 3: 21,039 ops/s

# Warmup Iteration 4: 20,919 ops/s

# Warmup Iteration 5: 20,912 ops/s

Iteration 1: 20,952 ops/s

Iteration 2: 20,822 ops/s

Iteration 3: 20,946 ops/s

Iteration 4: 20,916 ops/s

Iteration 5: 20,936 ops/s

# Run progress: 74,00% complete, ETA 00:22:04

# Fork: 3 of 5

# Warmup Iteration 1: 20,873 ops/s

# Warmup Iteration 2: 20,970 ops/s

# Warmup Iteration 3: 20,862 ops/s

# Warmup Iteration 4: 20,907 ops/s

# Warmup Iteration 5: 21,011 ops/s

Iteration 1: 20,963 ops/s

Iteration 2: 21,023 ops/s

Iteration 3: 20,927 ops/s

Iteration 4: 20,913 ops/s

Iteration 5: 20,878 ops/s

# Run progress: 76,00% complete, ETA 00:20:22

# Fork: 4 of 5

# Warmup Iteration 1: 20,823 ops/s

# Warmup Iteration 2: 20,956 ops/s

# Warmup Iteration 3: 20,973 ops/s

# Warmup Iteration 4: 21,046 ops/s

# Warmup Iteration 5: 20,815 ops/s

Iteration 1: 20,896 ops/s

Iteration 2: 20,896 ops/s

Iteration 3: 20,782 ops/s

Iteration 4: 20,880 ops/s

Iteration 5: 21,025 ops/s

# Run progress: 78,00% complete, ETA 00:18:40

# Fork: 5 of 5

# Warmup Iteration 1: 20,807 ops/s

# Warmup Iteration 2: 20,797 ops/s

# Warmup Iteration 3: 20,943 ops/s

# Warmup Iteration 4: 21,025 ops/s

# Warmup Iteration 5: 20,952 ops/s

Iteration 1: 21,049 ops/s

Iteration 2: 20,924 ops/s

Iteration 3: 21,009 ops/s

Iteration 4: 21,006 ops/s

Iteration 5: 21,034 ops/s

Result "main.com.StreamBenchmark.standardDeviationSequential":

20,942 ±(99.9%) 0,050 ops/s [Average]

(min, avg, max) = (20,782, 20,942, 21,049), stdev = 0,067

CI (99.9%): [20,892, 20,992] (assumes normal distribution)

# JMH version: 1.36

# VM version: JDK 21.0.1, OpenJDK 64-Bit Server VM, 21.0.1+12-29

# VM invoker: C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe

# VM options: -javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8

# Blackhole mode: compiler (auto-detected, use -Djmh.blackhole.autoDetect=false to disable)

# Warmup: 5 iterations, 10 s each

# Measurement: 5 iterations, 10 s each

# Timeout: 10 min per iteration

# Threads: 1 thread, will synchronize iterations

# Benchmark mode: Throughput, ops/time

# Benchmark: main.com.StreamBenchmark.sumParallel

# Run progress: 80,00% complete, ETA 00:16:58

# Fork: 1 of 5

# Warmup Iteration 1: 630,232 ops/s

# Warmup Iteration 2: 638,093 ops/s

# Warmup Iteration 3: 616,529 ops/s

# Warmup Iteration 4: 612,134 ops/s

# Warmup Iteration 5: 637,477 ops/s

Iteration 1: 634,169 ops/s

Iteration 2: 639,979 ops/s

Iteration 3: 625,742 ops/s

Iteration 4: 636,787 ops/s

Iteration 5: 635,516 ops/s

# Run progress: 82,00% complete, ETA 00:15:16

# Fork: 2 of 5

# Warmup Iteration 1: 610,749 ops/s

# Warmup Iteration 2: 637,803 ops/s

# Warmup Iteration 3: 646,253 ops/s

# Warmup Iteration 4: 635,189 ops/s

# Warmup Iteration 5: 611,989 ops/s

Iteration 1: 624,286 ops/s

Iteration 2: 644,849 ops/s

Iteration 3: 641,268 ops/s

Iteration 4: 625,714 ops/s

Iteration 5: 623,311 ops/s

# Run progress: 84,00% complete, ETA 00:13:34

# Fork: 3 of 5

# Warmup Iteration 1: 610,190 ops/s

# Warmup Iteration 2: 628,054 ops/s

# Warmup Iteration 3: 629,833 ops/s

# Warmup Iteration 4: 641,453 ops/s

# Warmup Iteration 5: 647,496 ops/s

Iteration 1: 637,499 ops/s

Iteration 2: 632,759 ops/s

Iteration 3: 644,736 ops/s

Iteration 4: 644,951 ops/s

Iteration 5: 631,592 ops/s

# Run progress: 86,00% complete, ETA 00:11:52

# Fork: 4 of 5

# Warmup Iteration 1: 628,167 ops/s

# Warmup Iteration 2: 640,515 ops/s

# Warmup Iteration 3: 639,783 ops/s

# Warmup Iteration 4: 629,062 ops/s

# Warmup Iteration 5: 638,217 ops/s

Iteration 1: 633,048 ops/s

Iteration 2: 637,744 ops/s

Iteration 3: 620,078 ops/s

Iteration 4: 633,925 ops/s

Iteration 5: 633,806 ops/s

# Run progress: 88,00% complete, ETA 00:10:10

# Fork: 5 of 5

# Warmup Iteration 1: 619,859 ops/s

# Warmup Iteration 2: 631,729 ops/s

# Warmup Iteration 3: 645,817 ops/s

# Warmup Iteration 4: 641,686 ops/s

# Warmup Iteration 5: 638,280 ops/s

Iteration 1: 638,955 ops/s

Iteration 2: 648,389 ops/s

Iteration 3: 646,221 ops/s

Iteration 4: 643,664 ops/s

Iteration 5: 630,767 ops/s

Result "main.com.StreamBenchmark.sumParallel":

635,590 ±(99.9%) 5,816 ops/s [Average]

(min, avg, max) = (620,078, 635,590, 648,389), stdev = 7,765

CI (99.9%): [629,774, 641,407] (assumes normal distribution)

# JMH version: 1.36

# VM version: JDK 21.0.1, OpenJDK 64-Bit Server VM, 21.0.1+12-29

# VM invoker: C:\Users\User\.jdks\openjdk-21.0.1\bin\java.exe

# VM options: -javaagent:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea\_rt.jar=62534:C:\Users\User\AppData\Local\Programs\IntelliJ IDEA Ultimate\bin -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8

# Blackhole mode: compiler (auto-detected, use -Djmh.blackhole.autoDetect=false to disable)

# Warmup: 5 iterations, 10 s each

# Measurement: 5 iterations, 10 s each

# Timeout: 10 min per iteration

# Threads: 1 thread, will synchronize iterations

# Benchmark mode: Throughput, ops/time

# Benchmark: main.com.StreamBenchmark.sumSequential

# Run progress: 90,00% complete, ETA 00:08:28

# Fork: 1 of 5

# Warmup Iteration 1: 142,078 ops/s

# Warmup Iteration 2: 142,001 ops/s

# Warmup Iteration 3: 141,652 ops/s

# Warmup Iteration 4: 146,543 ops/s

# Warmup Iteration 5: 156,820 ops/s

Iteration 1: 155,788 ops/s

Iteration 2: 155,310 ops/s

Iteration 3: 156,593 ops/s

Iteration 4: 156,703 ops/s

Iteration 5: 156,576 ops/s

# Run progress: 92,00% complete, ETA 00:06:47

# Fork: 2 of 5

# Warmup Iteration 1: 143,058 ops/s

# Warmup Iteration 2: 142,428 ops/s

# Warmup Iteration 3: 142,723 ops/s

# Warmup Iteration 4: 144,269 ops/s

# Warmup Iteration 5: 156,194 ops/s

Iteration 1: 157,139 ops/s

Iteration 2: 158,010 ops/s

Iteration 3: 157,546 ops/s

Iteration 4: 156,583 ops/s

Iteration 5: 157,126 ops/s

# Run progress: 94,00% complete, ETA 00:05:05

# Fork: 3 of 5

# Warmup Iteration 1: 145,929 ops/s

# Warmup Iteration 2: 145,708 ops/s

# Warmup Iteration 3: 146,176 ops/s

# Warmup Iteration 4: 150,637 ops/s

# Warmup Iteration 5: 155,707 ops/s

Iteration 1: 155,947 ops/s

Iteration 2: 155,161 ops/s

Iteration 3: 155,102 ops/s

Iteration 4: 156,681 ops/s

Iteration 5: 156,202 ops/s

# Run progress: 96,00% complete, ETA 00:03:23

# Fork: 4 of 5

# Warmup Iteration 1: 144,157 ops/s

# Warmup Iteration 2: 144,988 ops/s

# Warmup Iteration 3: 144,921 ops/s

# Warmup Iteration 4: 149,019 ops/s

# Warmup Iteration 5: 157,552 ops/s

Iteration 1: 156,405 ops/s

Iteration 2: 156,374 ops/s

Iteration 3: 148,778 ops/s

Iteration 4: 157,024 ops/s

Iteration 5: 156,560 ops/s

# Run progress: 98,00% complete, ETA 00:01:41

# Fork: 5 of 5

# Warmup Iteration 1: 143,131 ops/s

# Warmup Iteration 2: 144,028 ops/s

# Warmup Iteration 3: 143,389 ops/s

# Warmup Iteration 4: 147,188 ops/s

# Warmup Iteration 5: 156,613 ops/s

Iteration 1: 155,462 ops/s

Iteration 2: 156,305 ops/s

Iteration 3: 157,029 ops/s

Iteration 4: 156,416 ops/s

Iteration 5: 156,203 ops/s

Result "main.com.StreamBenchmark.sumSequential":

156,121 ±(99.9%) 1,262 ops/s [Average]

(min, avg, max) = (148,778, 156,121, 158,010), stdev = 1,685

CI (99.9%): [154,859, 157,383] (assumes normal distribution)

# Run complete. Total time: 01:24:45

REMEMBER: The numbers below are just data. To gain reusable insights, you need to follow up on

why the numbers are the way they are. Use profilers (see -prof, -lprof), design factorial

experiments, perform baseline and negative tests that provide experimental control, make sure

the benchmarking environment is safe on JVM/OS/HW level, ask for reviews from the domain experts.

Do not assume the numbers tell you what you want them to tell.

NOTE: Current JVM experimentally supports Compiler Blackholes, and they are in use. Please exercise

extra caution when trusting the results, look into the generated code to check the benchmark still

works, and factor in a small probability of new VM bugs. Additionally, while comparisons between

different JVMs are already problematic, the performance difference caused by different Blackhole

modes can be very significant. Please make sure you use the consistent Blackhole mode for comparisons.

Benchmark Mode Cnt Score Error Units

StreamBenchmark.averageParallel thrpt 25 567,370 ± 27,651 ops/s

StreamBenchmark.averageSequential thrpt 25 90,822 ± 1,387 ops/s

StreamBenchmark.filterParallel thrpt 25 38,244 ± 0,725 ops/s

StreamBenchmark.filterSequential thrpt 25 12,328 ± 0,050 ops/s

StreamBenchmark.multiplyParallel thrpt 25 2,273 ± 0,097 ops/s

StreamBenchmark.multiplySequential thrpt 25 4,094 ± 0,153 ops/s

StreamBenchmark.standardDeviationParallel thrpt 25 194,967 ± 1,913 ops/s

StreamBenchmark.standardDeviationSequential thrpt 25 20,942 ± 0,050 ops/s

StreamBenchmark.sumParallel thrpt 25 635,590 ± 5,816 ops/s

StreamBenchmark.sumSequential thrpt 25 156,121 ± 1,262 ops/s

Process finished with exit code 0