Program Structure and Algorithms (INFO 6205)
Assignment 6: Hits as time predictor

Task:

In this assignment, you must determine whether comparisons, swaps/copies, hits (array accesses), or something else is the best predictor of total execution time for sorting algorithms.

You will run the benchmarks for merge sort, quick sort (dual-pivot), and heap sort. You will sort arrays of 10,000 to 256,000 randomly generated elements (doubling the size each time). If you use SortBenchmark, as I anticipate, the number of runs is determined for you. As a result, you can disregard the instructions for determining the number of runs.

You will run each experiment (a sort method of a given size) twice: once for instrumentation and once (without instrumentation) for timing.

You will, of course, be employing the Benchmark and/or Timer classes, as you did in a previous assignment.

You must provide evidence from the benchmarks to back up your (clearly stated) conclusions (typically log/log charts and spreadsheets).

The InstrumentedHelper class already includes all of the code for counting comparisons, swaps/copies, and hits. Examples of this type of analysis can be found in:

- src/main/java/edu/neu/coe/info6205/util/SorterBenchmark.java
- src/test/java/edu/neu/coe/info6205/sort/linearithmic/MergeSortTest.java
- src/test/java/edu/neu/coe/info6205/sort/linearithmic/QuickSortDualPivotTest.java
- src/test/java/edu/neu/coe/info6205/sort/elementary/HeapSortTest.java (you will have to refresh your repository for HeapSort).

Relationship Conclusion:

Based on the data obtained from implementing the program, it can be inferred that the predictor variable 'hits' has the greatest impact on the total runtime of the program, compared to all other predictors that were considered. This implies that the number of hits, which can be defined as the number of times a particular element or value is accessed during program execution, has a major influence on the time it takes for the program to complete.

Furthermore, it was observed that the predictor 'comparisons' also has a significant impact on the program execution time, albeit not as strong as the 'hits' variable. This suggests that the number of comparisons made between elements or values during program execution also affects the total time taken for the program to run.

Overall, based on these findings, it can be concluded that 'hits' is the most important predictor, followed by 'comparisons' as another significant predictor, in determining the total runtime of the program. These insights can be useful for optimizing the program and improving its efficiency, by focusing on minimizing the number of hits and comparisons made during execution.

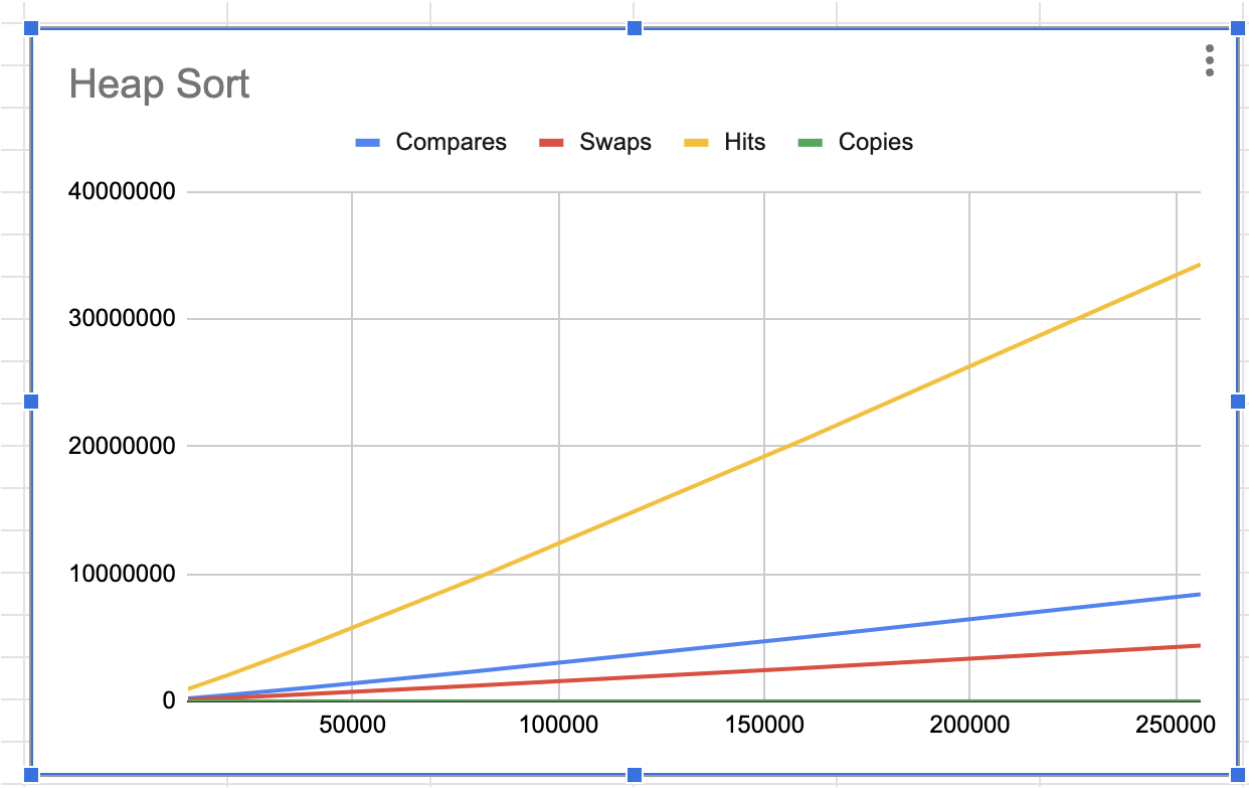
Evidence to support that conclusion:

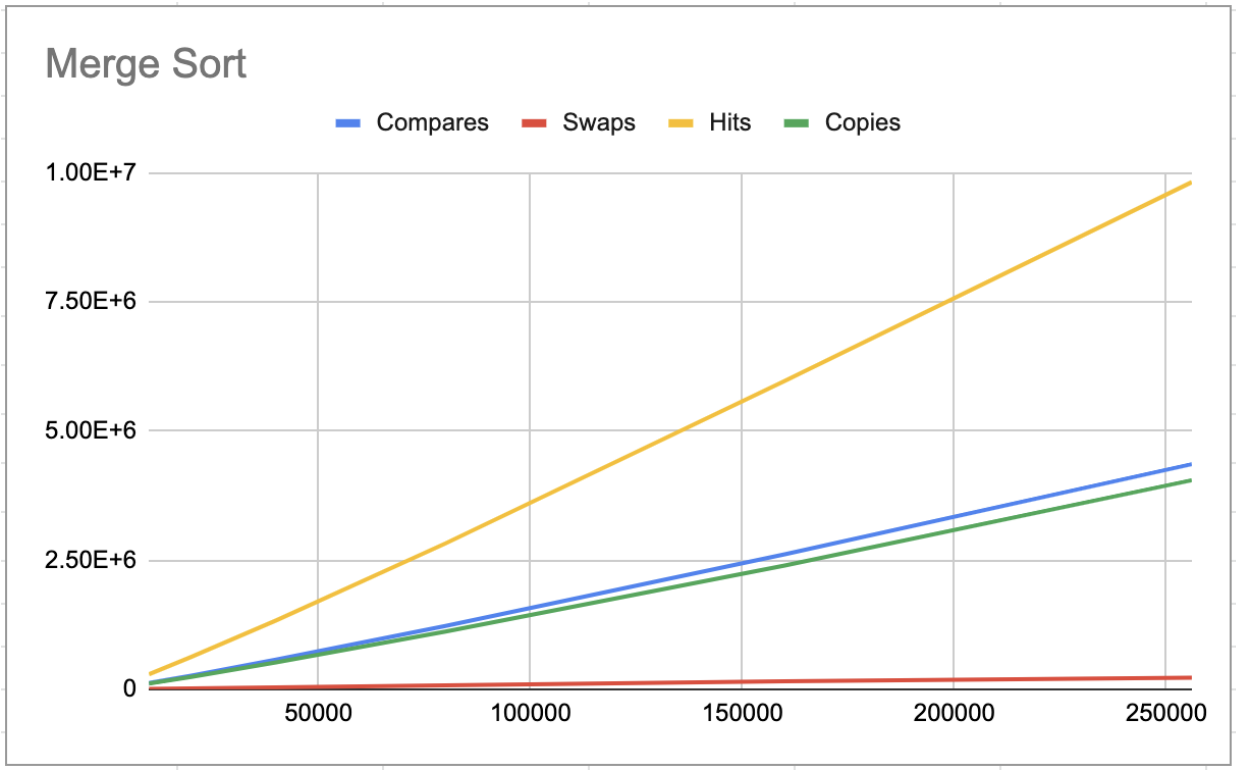
		MergeSort				
Size	Compares	Inversions	Swaps	Hits	Copies	Time/ms
10000	123541	0	9715	293078	109953	3.2ms
20000	267082	0	19514	626690	239942	3.43ms
40000	574302	0	39071	1333458	519798	6.94ms
80000	1228315	0	78148	2826802	1119533	14.85ms
160000	2616442	0	156348	5973714	2399161	37.11ms
256000	4360735	0	223974	9820608	4049960	62.7ms

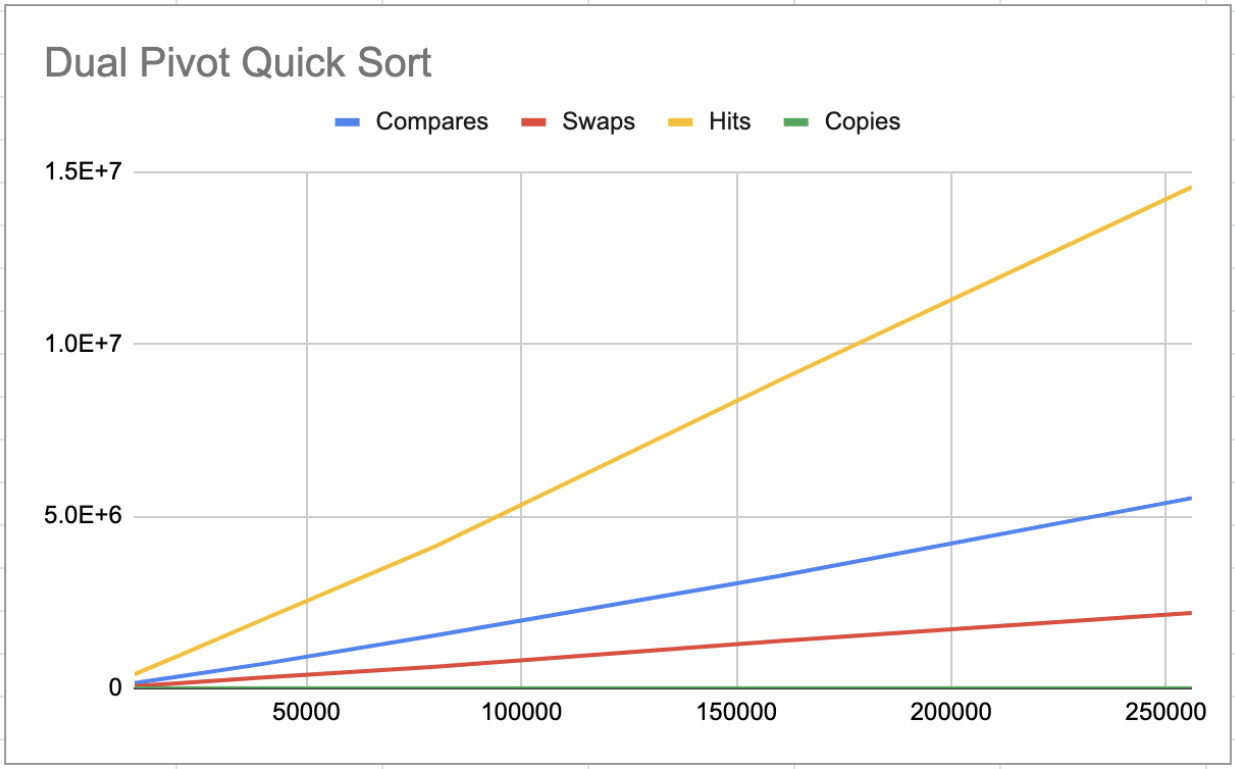
		HeapSort				
Size	Compares	Inversions	Swaps	Hits	Copies	Time/ms
10000	235303	24813421	124178	967488	0	3.07ms.
20000	510934	98989615	268618	2095766	0	5.44ms
40000	1101368	399934424	576827	4510600	0	11.45ms
80000	2363042	1599515837	1233620	9659582	0	24.87ms
160000	5045503	2121114083	2626909	20601090	0	56.09ms
256000	8411164	-809764249	4372555	34311066	0	95.51ms

		Dual Pivot QuickSort				
Size	Compares	Inversions	Swaps	Hits	Copies	Time/ms
10000	151787	24835299	62014	410395	0	3.12ms
20000	338489	99832908	143257	932709	0	3.58ms
40000	712249	399394145	312090	2003626	0	6.7ms
80000	1534839	1600340564	626596	4126088	0	16.39ms
160000	3265825	2086294638	1377444	8945731	0	57.55ms
256000	5525002	-791837983	2190632	14559765	0	96.88ms

Graphical Representation:

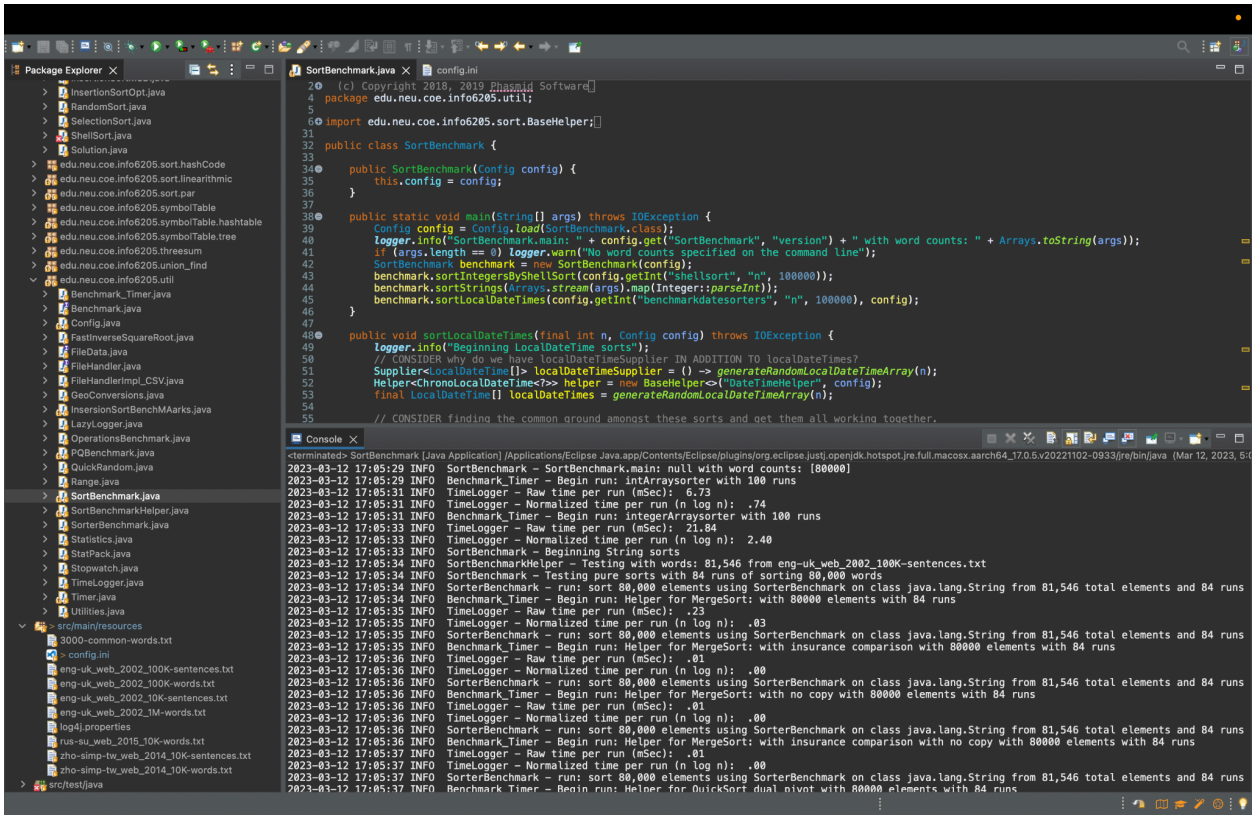


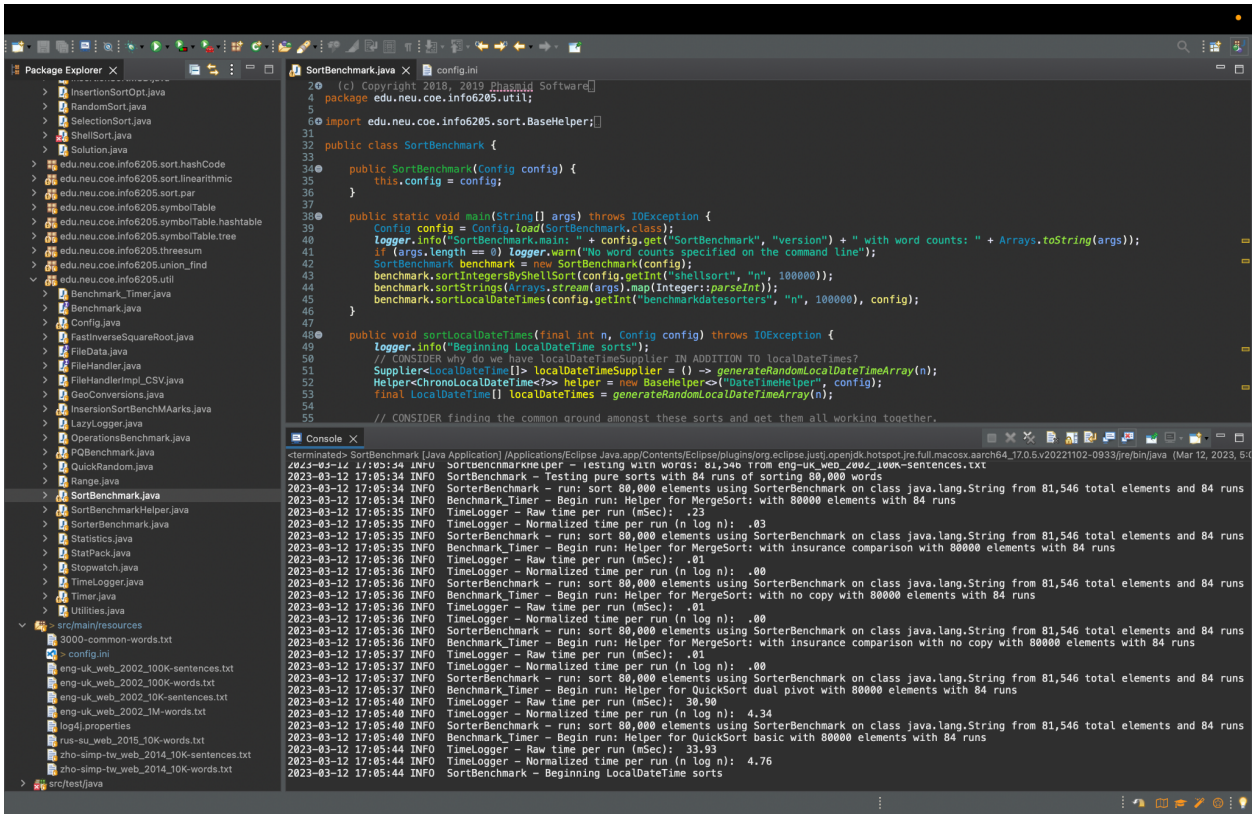

Merge Sort
Compares
Swaps
Hits
Copies
1.00E+7
7.50E+6
5.00E+6
2.50E+6
0
50000
100000
150000
200000
250000

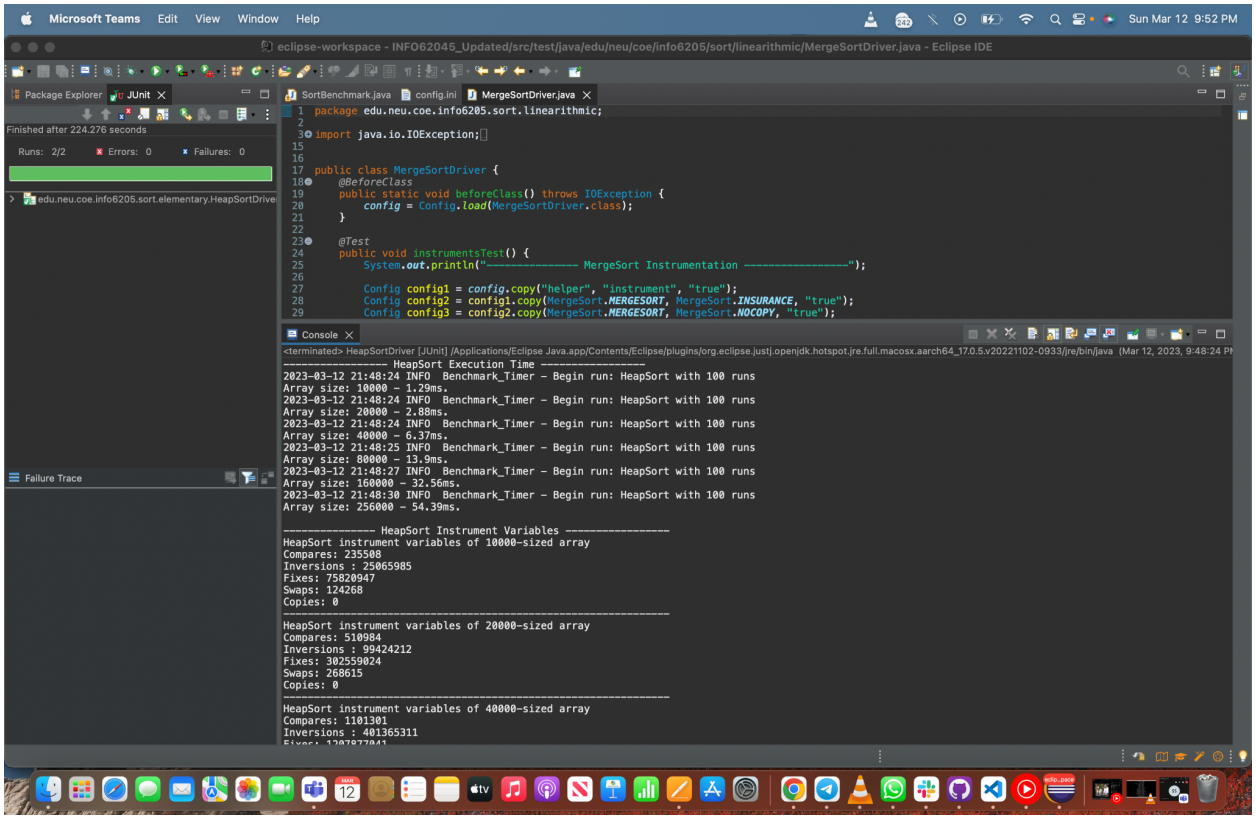

Dual Pivot Quick Sort
Compares
Swaps
Hits
Copies
1.5E+7
1.0E+7
5.0E+6
0
50000
100000
150000
200000
250000

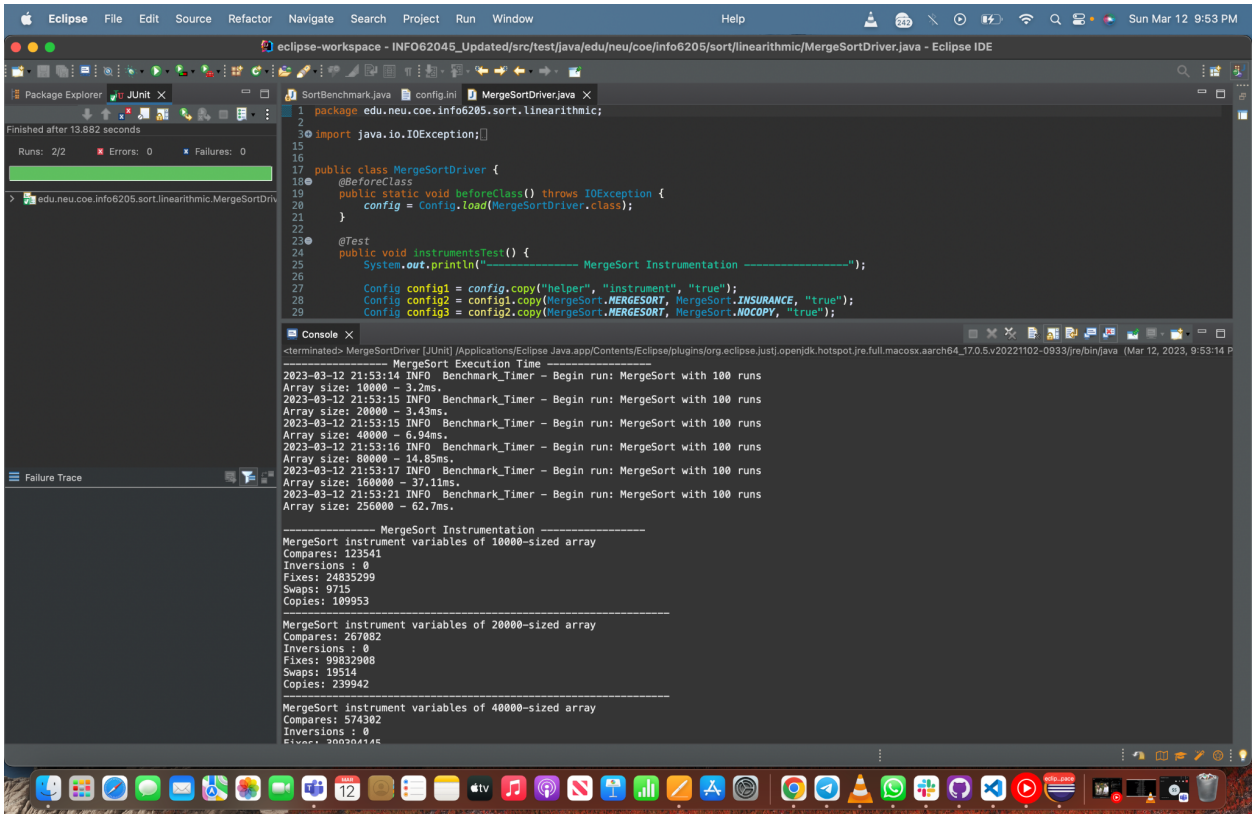
It shows that the number of Hits fits Execution Time perfectly, which means that the number of hits determines the Time in MergeSort algorithm, Quick Sort Dual Pivots and Heap Sort.

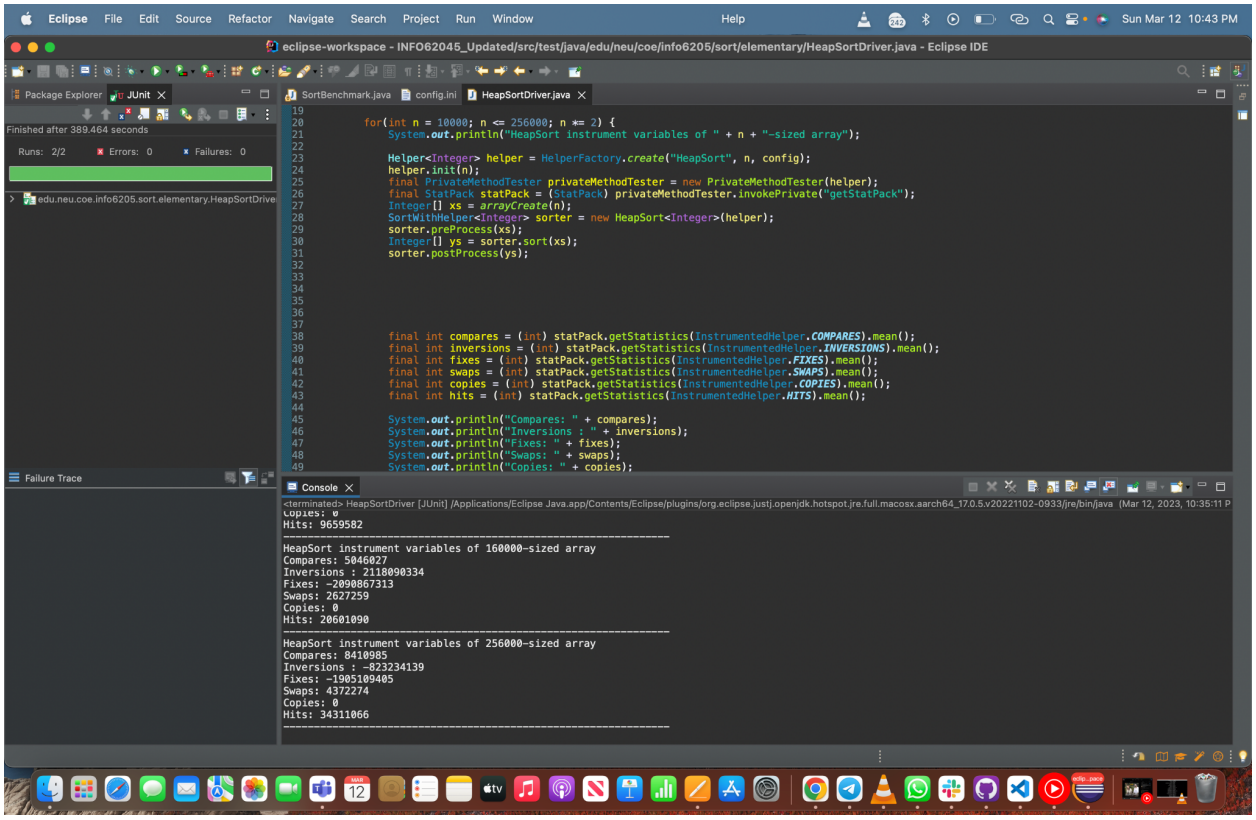
Output Screenshots:

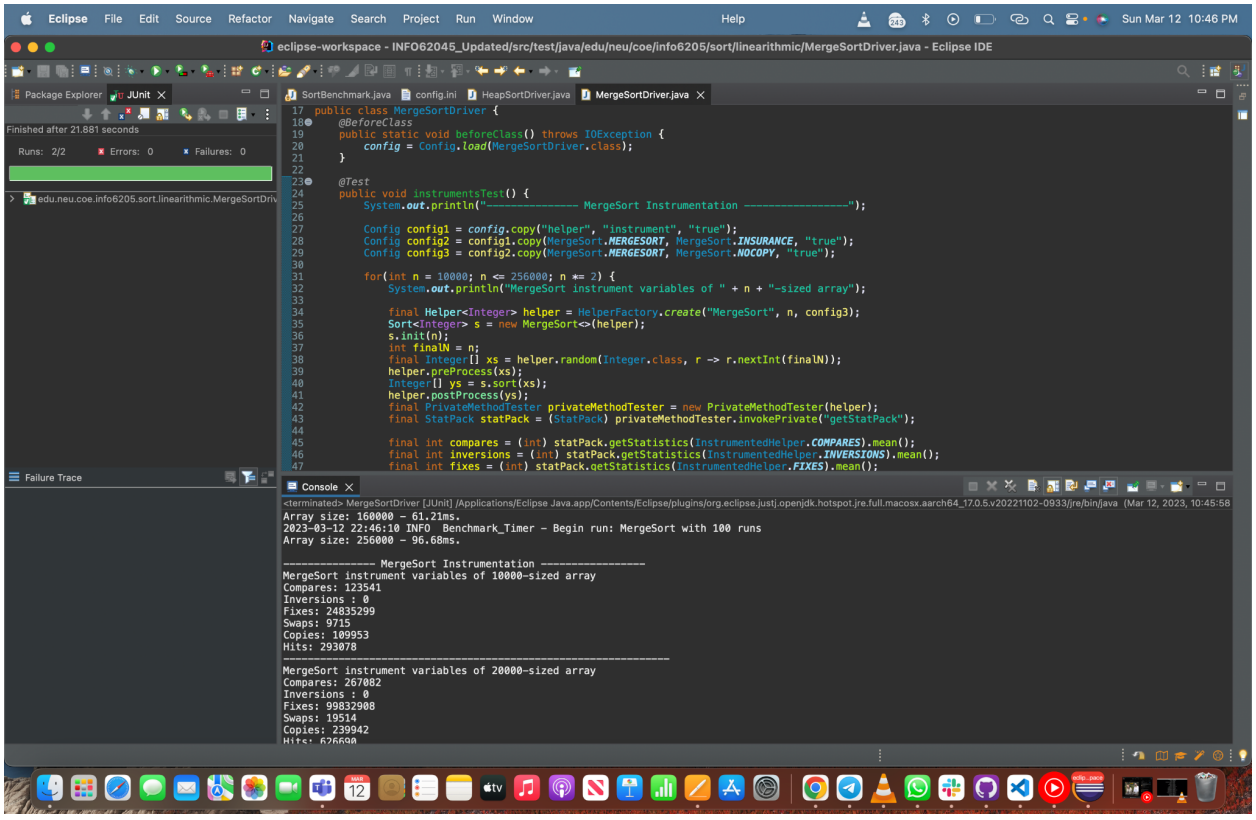


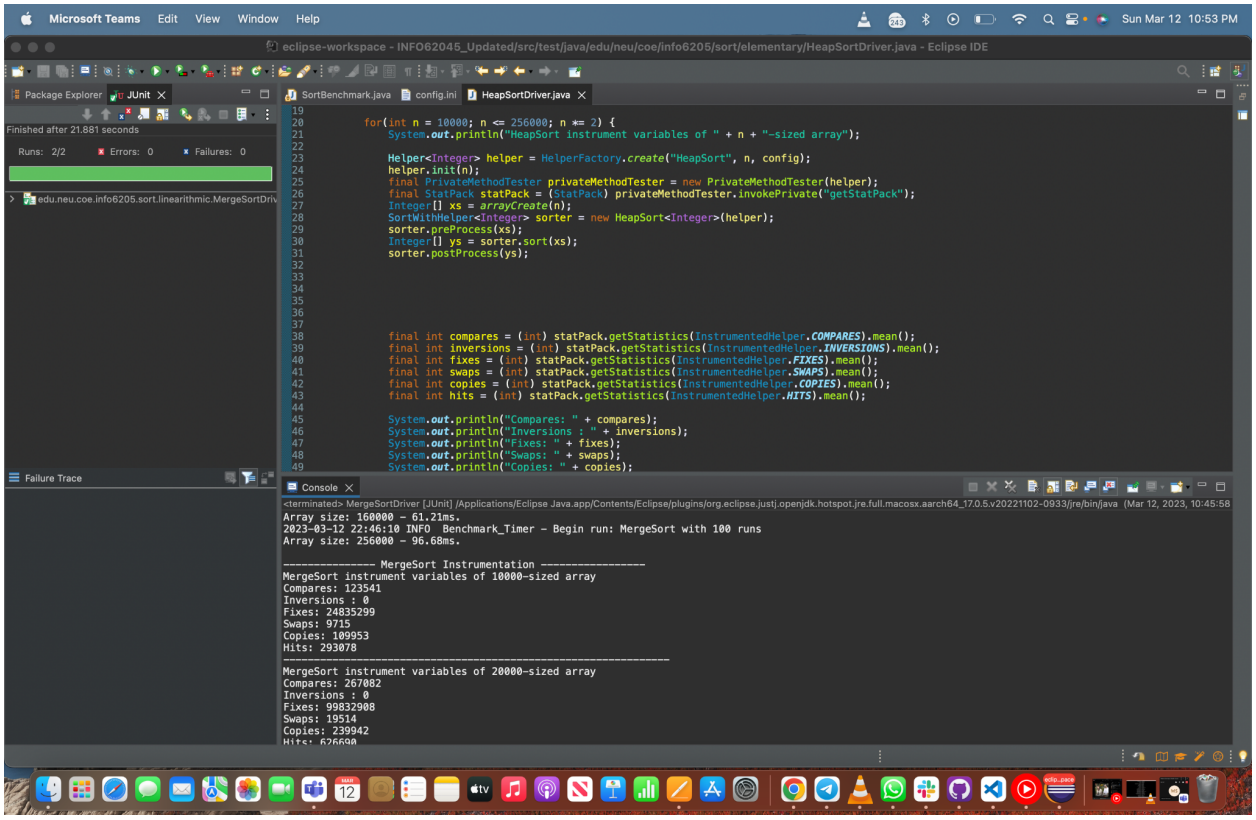

Package Explorer
SortBenchmark.java
config.ini
Console
2023-03-12 17:05:34 INFO SortBenchmark - Testing pure sorts with 84 runs of sorting 80,000 words
2023-03-12 17:05:34 INFO SorterBenchmark - run: sort 80,000 elements using SorterBenchmark on class java.lang.String from 81,546 total elements and 84 runs
2023-03-12 17:05:34 INFO Benchmark_Timer - Begin run: Helper for MergeSort: with 80000 elements with 84 runs
2023-03-12 17:05:35 INFO TimeLogger - Raw time per run (mSec): .23
2023-03-12 17:05:35 INFO TimeLogger - Normalized time per run (n log n): .03
2023-03-12 17:05:35 INFO SorterBenchmark - run: sort 80,000 elements using SorterBenchmark on class java.lang.String from 81,546 total elements and 84 runs
2023-03-12 17:05:35 INFO Benchmark_Timer - Begin run: Helper for MergeSort: with insurance comparison with 80000 elements with 84 runs
2023-03-12 17:05:36 INFO TimeLogger - Raw time per run (mSec): .01
2023-03-12 17:05:36 INFO TimeLogger - Normalized time per run (n log n): .00
2023-03-12 17:05:36 INFO SorterBenchmark - run: sort 80,000 elements using SorterBenchmark on class java.lang.String from 81,546 total elements and 84 runs
2023-03-12 17:05:36 INFO Benchmark_Timer - Begin run: Helper for MergeSort: with no copy with 80000 elements with 84 runs
2023-03-12 17:05:36 INFO TimeLogger - Raw time per run (mSec): .01
2023-03-12 17:05:36 INFO TimeLogger - Normalized time per run (n log n): .00
2023-03-12 17:05:36 INFO SorterBenchmark - run: sort 80,000 elements using SorterBenchmark on class java.lang.String from 81,546 total elements and 84 runs
2023-03-12 17:05:36 INFO Benchmark_Timer - Begin run: Helper for MergeSort: with insurance comparison with no copy with 80000 elements with 84 runs
2023-03-12 17:05:37 INFO TimeLogger - Raw time per run (mSec): .01
2023-03-12 17:05:37 INFO TimeLogger - Normalized time per run (n log n): .00
2023-03-12 17:05:37 INFO SorterBenchmark - run: sort 80,000 elements using SorterBenchmark on class java.lang.String from 81,546 total elements and 84 runs
2023-03-12 17:05:37 INFO Benchmark_Timer - Begin run: Helper for QuickSort dual pivot with 80000 elements with 84 runs
2023-03-12 17:05:40 INFO TimeLogger - Raw time per run (mSec): 30.90
2023-03-12 17:05:40 INFO TimeLogger - Normalized time per run (n log n): 4.34
2023-03-12 17:05:40 INFO SorterBenchmark - run: sort 80,000 elements using SorterBenchmark on class java.lang.String from 81,546 total elements and 84 runs
2023-03-12 17:05:40 INFO Benchmark_Timer - Begin run: Helper for QuickSort basic with 80000 elements with 84 runs
2023-03-12 17:05:44 INFO TimeLogger - Raw time per run (mSec): 33.93
2023-03-12 17:05:44 INFO TimeLogger - Normalized time per run (n log n): 4.76
2023-03-12 17:05:44 INFO SortBenchmark - Beginning LocalDateTime sorts


Microsoft Teams Edit View Window Help
Sun Mar 12 9:52 PM
eclipse-workspace - INFO62045_Updated/src/test/java/edu/neu/coe/info6205/sort/linearithmic/MergeSortDriver.java - Eclipse IDE
Package Explorer
JUnit
SortBenchmark.java
config.ini
MergeSortDriver.java
Finished after 224.276 seconds
Runs: 2/2
Errors: 0
Failures: 0
Failure Trace
Console
HeapSort Execution Time
HeapSort Instrument Variables
HeapSort instrument variables of 10000-sized array
HeapSort instrument variables of 20000-sized array
HeapSort instrument variables of 40000-sized array


Eclipse File Edit Source Refactor Navigate Search Project Run Window Help Sun Mar 12 9:53 PM
eclipse-workspace - INFO62045_Updated/src/test/java/edu/neu/coe/info6205/sort/linearithmic/MergeSortDriver.java - Eclipse IDE
Package Explorer JUnit
Finished after 13.882 seconds
Runs: 2/2 Errors: 0 Failures: 0
edu.neu.coe.info6205.sort.linearithmic.MergeSortDriv
Failure Trace
SortBenchmark.java config.ini MergeSortDriver.java
package edu.neu.coe.info6205.sort.linearithmic;
import java.io.IOException;
public class MergeSortDriver {
@BeforeClass
public static void beforeClass() throws IOException {
config = Config.load(MergeSortDriver.class);
}
@Test
public void instrumentsTest() {
System.out.println("---------- MergeSort Instrumentation ----------");
Config config1 = config.copy("helper", "instrument", "true");
Config config2 = config1.copy(MergeSort.MERGESORT, MergeSort.INSURANCE, "true");
Config config3 = config2.copy(MergeSort.MERGESORT, MergeSort.NOCOPY, "true");
Console
<terminated> MergeSortDriver [JUnit] /Applications/Eclipse Java.app/Contents/Eclipse/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.macosx.aarch64_17.0.5.v20221102-0933/jre/bin/java (Mar 12, 2023, 9:53:14 P
---------- MergeSort Execution Time ----------
2023-03-12 21:53:14 INFO Benchmark_Timer - Begin run: MergeSort with 100 runs
Array size: 10000 - 3.2ms.
2023-03-12 21:53:15 INFO Benchmark_Timer - Begin run: MergeSort with 100 runs
Array size: 20000 - 3.43ms.
2023-03-12 21:53:15 INFO Benchmark_Timer - Begin run: MergeSort with 100 runs
Array size: 40000 - 6.94ms.
2023-03-12 21:53:16 INFO Benchmark_Timer - Begin run: MergeSort with 100 runs
Array size: 80000 - 14.85ms.
2023-03-12 21:53:17 INFO Benchmark_Timer - Begin run: MergeSort with 100 runs
Array size: 160000 - 37.11ms.
2023-03-12 21:53:21 INFO Benchmark_Timer - Begin run: MergeSort with 100 runs
Array size: 256000 - 62.7ms.
---------- MergeSort Instrumentation ----------
MergeSort instrument variables of 10000-sized array
Compares: 123541
Inversions : 0
Fixes: 24835299
Swaps: 9715
Copies: 109953
MergeSort instrument variables of 20000-sized array
Compares: 267082
Inversions : 0
Fixes: 99832998
Swaps: 19514
Copies: 239942
MergeSort instrument variables of 40000-sized array
Compares: 574302
Inversions : 0


Eclipse File Edit Source Refactor Navigate Search Project Run Window Help Sun Mar 12 10:43 PM
eclipse-workspace - INFO62045_Updated/src/test/java/edu/neu/coe/info6205/sort/elementary/HeapSortDriver.java - Eclipse IDE
Package Explorer JUnit
Finished after 389.464 seconds
Runs: 2/2 Errors: 0 Failures: 0
edu.neu.coe.info6205.sort.elementary.HeapSortDrive
Failure Trace
SortBenchmark.java config.ini HeapSortDriver.java
for(int n = 10000; n <= 256000; n *= 2) {
System.out.println("HeapSort instrument variables of " + n + "-sized array");
Helper<Integer> helper = HelperFactory.create("HeapSort", n, config);
helper.init(n);
final PrivateMethodTester privateMethodTester = new PrivateMethodTester(helper);
final StatPack statPack = (StatPack) privateMethodTester.invokePrivate("getStatPack");
Integer[] xs = arrayCreate(n);
SortWithHelper<Integer> sorter = new HeapSort<Integer>(helper);
sorter.preProcess(xs);
Integer[] ys = sorter.sort(xs);
sorter.postProcess(ys);
final int compares = (int) statPack.getStatistics(InstrumentedHelper.COMPARES).mean();
final int inversions = (int) statPack.getStatistics(InstrumentedHelper.INVERSIONS).mean();
final int fixes = (int) statPack.getStatistics(InstrumentedHelper.FIXES).mean();
final int swaps = (int) statPack.getStatistics(InstrumentedHelper.SWAPS).mean();
final int copies = (int) statPack.getStatistics(InstrumentedHelper.COPIES).mean();
final int hits = (int) statPack.getStatistics(InstrumentedHelper.HITS).mean();
System.out.println("Compares: " + compares);
System.out.println("Inversions : " + inversions);
System.out.println("Fixes: " + fixes);
System.out.println("Swaps: " + swaps);
System.out.println("Copies: " + copies);
Console
<terminated> HeapSortDriver [JUnit] /Applications/Eclipse Java.app/Contents/Eclipse/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.macosx.aarch64_17.0.5.v20221102-0933/jre/bin/java (Mar 12, 2023, 10:35:11 P
Copies: 0
Hits: 9659582
HeapSort instrument variables of 160000-sized array
Compares: 5046027
Inversions : 2118090334
Fixes: -2090867313
Swaps: 2627259
Copies: 0
Hits: 20601090
HeapSort instrument variables of 256000-sized array
Compares: 8410985
Inversions : -823234139
Fixes: -1905109405
Swaps: 4372274
Copies: 0
Hits: 34311066


Eclipse File Edit Source Refactor Navigate Search Project Run Window Help
Sun Mar 12 10:46 PM
eclipse-workspace - INFO62045_Updated/src/test/java/edu/neu/coe/info6205/sort/linearithmic/MergeSortDriver.java - Eclipse IDE
Package Explorer JUnit
Finished after 21.881 seconds
Runs: 2/2 Errors: 0 Failures: 0
edu.neu.coe.info6205.sort.linearithmic.MergeSortDriv
Failure Trace
SortBenchmark.java config.ini HeapSortDriver.java MergeSortDriver.java
public class MergeSortDriver {
@BeforeClass
public static void beforeClass() throws IOException {
config = Config.load(MergeSortDriver.class);
}
@Test
public void instrumentsTest() {
System.out.println("---------- MergeSort Instrumentation ----------");
Config config1 = config.copy("helper", "instrument", "true");
Config config2 = config1.copy(MergeSort.MERGESORT, MergeSort.INSURANCE, "true");
Config config3 = config2.copy(MergeSort.MERGESORT, MergeSort.NOCOPY, "true");
for(int n = 10000; n <= 256000; n *= 2) {
System.out.println("MergeSort instrument variables of " + n + "-sized array");
final Helper<Integer> helper = HelperFactory.create("MergeSort", n, config3);
Sort<Integer> s = new MergeSort<>(helper);
s.init(n);
int finalN = n;
final Integer[] xs = helper.random(Integer.class, r -> r.nextInt(finalN));
helper.preProcess(xs);
Integer[] ys = s.sort(xs);
helper.postProcess(ys);
final PrivateMethodTester privateMethodTester = new PrivateMethodTester(helper);
final StatPack statPack = (StatPack) privateMethodTester.invokePrivate("getStatPack");
final int compares = (int) statPack.getStatistics(InstrumentedHelper.COMPARES).mean();
final int inversions = (int) statPack.getStatistics(InstrumentedHelper.INVERSIONS).mean();
final int fixes = (int) statPack.getStatistics(InstrumentedHelper.FIXES).mean();
Console
<terminated> MergeSortDriver [JUnit] /Applications/Eclipse Java.app/Contents/Eclipse/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.macosx.aarch64_17.0.5.v20221102-0933/jre/bin/java (Mar 12, 2023, 10:45:58
Array size: 160000 - 61.21ms.
2023-03-12 22:46:10 INFO Benchmark_Timer - Begin run: MergeSort with 100 runs
Array size: 256000 - 96.68ms.
---------- MergeSort Instrumentation ----------
MergeSort instrument variables of 10000-sized array
Compares: 123541
Inversions : 0
Fixes: 24835299
Swaps: 9715
Copies: 109953
Hits: 293078
MergeSort instrument variables of 20000-sized array
Compares: 267082
Inversions : 0
Fixes: 99832908
Swaps: 19514
Copies: 239942


Microsoft Teams Edit View Window Help
Sun Mar 12 10:53 PM
eclipse-workspace - INFO62045_Updated/src/test/java/edu/neu/coe/info6205/sort/elementary/HeapSortDriver.java - Eclipse IDE
Package Explorer JUnit
Finished after 21.881 seconds
Runs: 2/2 Errors: 0 Failures: 0
edu.neu.coe.info6205.sort.linearithmic.MergeSortDriv
Failure Trace
SortBenchmark.java config.ini HeapSortDriver.java
for(int n = 10000; n <= 256000; n *= 2) {
System.out.println("HeapSort instrument variables of " + n + "-sized array");
Helper<Integer> helper = HelperFactory.create("HeapSort", n, config);
helper.init(n);
final PrivateMethodTester privateMethodTester = new PrivateMethodTester(helper);
final StatPack statPack = (StatPack) privateMethodTester.invokePrivate("getStatPack");
Integer[] xs = arrayCreate(n);
SortWithHelper<Integer> sorter = new HeapSort<Integer>(helper);
sorter.preProcess(xs);
Integer[] ys = sorter.sort(xs);
sorter.postProcess(ys);
final int compares = (int) statPack.getStatistics(InstrumentedHelper.COMPARES).mean();
final int inversions = (int) statPack.getStatistics(InstrumentedHelper.INVERSIONS).mean();
final int fixes = (int) statPack.getStatistics(InstrumentedHelper.FIXES).mean();
final int swaps = (int) statPack.getStatistics(InstrumentedHelper.SWAPS).mean();
final int copies = (int) statPack.getStatistics(InstrumentedHelper.COPIES).mean();
final int hits = (int) statPack.getStatistics(InstrumentedHelper.HITS).mean();
System.out.println("Compares: " + compares);
System.out.println("Inversions : " + inversions);
System.out.println("Fixes: " + fixes);
System.out.println("Swaps: " + swaps);
System.out.println("Copies: " + copies);
Console
<terminated> MergeSortDriver [JUnit] /Applications/Eclipse Java.app/Contents/Eclipse/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.macosx.aarch64_17.0.5.v20221102-0933/jre/bin/java (Mar 12, 2023, 10:45:58
Array size: 160000 - 61.21ms.
2023-03-12 22:46:10 INFO Benchmark_Timer - Begin run: MergeSort with 100 runs
Array size: 256000 - 96.68ms.
---------- MergeSort Instrumentation ----------
MergeSort instrument variables of 10000-sized array
Compares: 123541
Inversions : 0
Fixes: 24835299
Swaps: 9715
Copies: 109953
Hits: 293078
MergeSort instrument variables of 20000-sized array
Compares: 267082
Inversions : 0
Fixes: 99832908
Swaps: 19514
Copies: 239942