INFO 6205-PROGRAM STRUCTURE AND ALGORITHMS ASSIGNMENT-6

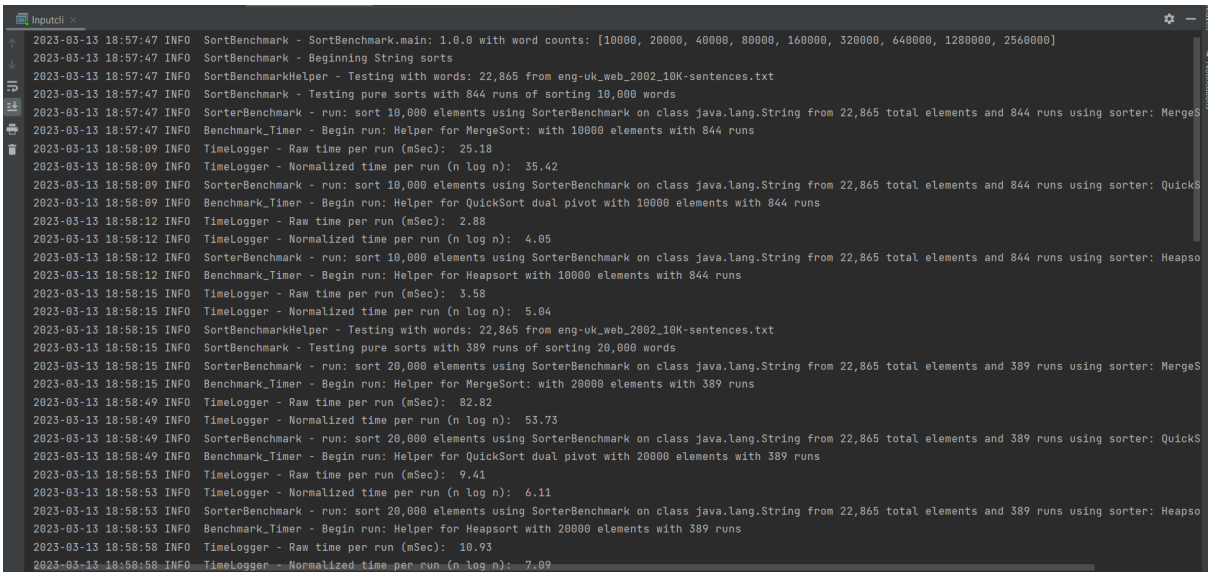
Sai Kashyap Cheruku
NU-ID- 002756849

Task:
The task is to identify the most accurate predictor of overall algorithm execution time.(comparisons, swaps/copies or array accesses)

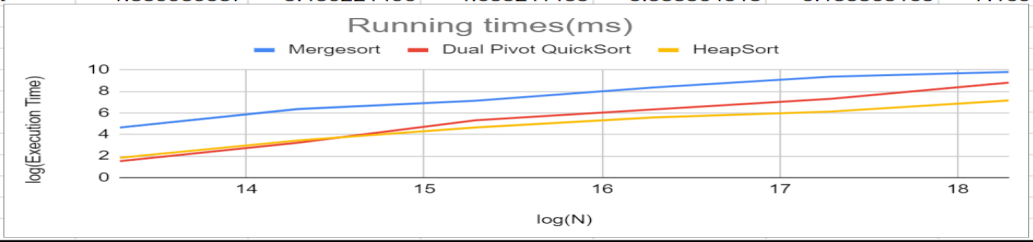
Relationship Conclusion:
We can infer from the graphs and data that hits and compares are both significant indicators of how long a sorting algorithm will take to execute. Using swaps as an indicator might not be advisable, as the traditional mergesort algorithm does not employ swaps to reach at a sorted solution.

Graphical Interpretations:

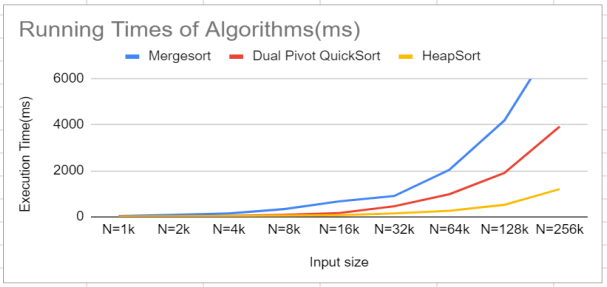
Without Instrumentation



	INPUT SIZE					
	13.28771238	14.28771238	15.28771238	16.28771238	17.28771238	18.28771238
Mergesort	4.654206378	6.371907298	7.141902767	8.375691104	9.38383387	9.815895594
Dual Pivot Quick	1.526068812	3.234194723	5.3305584	6.333960351	7.327687364	8.831497268
HeapSort	1.839959587	3.450221496	4.658211483	5.588864518	6.135863165	7.169123281



	N=1k	N=2k	N=4k	N=8k	N=16k	N=32k	N=64k	N=128k	N=256k
Mergesort	25.18	82.82	141.23	332.15	668.06	901.32	2044.65	4200.21	8022.98
Dual Pivot Quick	2.88	9.41	40.24	80.67	160.64	455.56	981.85	1907.99	3929.01
HeapSort	3.58	10.93	25.25	48.13	70.32	143.92	259.28	520.33	1203.26



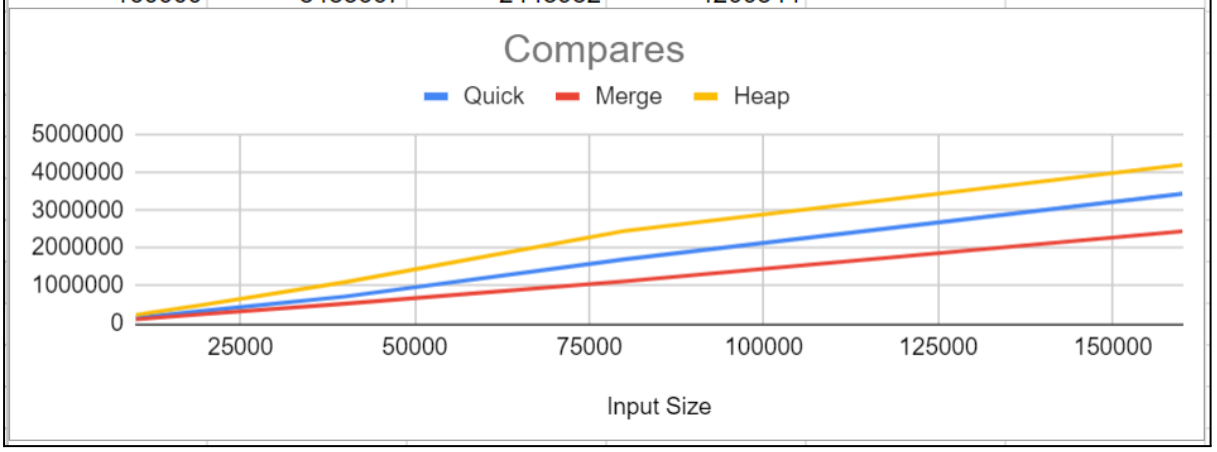
With Instrumentation

```
"C:\Program Files\Java\jdk-19\bin\java.exe" ...
2023-03-13 20:29:26 INFO  SortBenchmark - SortBenchmark.main: 1.0.0 with word counts: [10000, 20000, 40000, 80000, 160000, 320000, 640000, 1280000, 2560000]
2023-03-13 20:29:26 INFO  SortBenchmark - Beginning String sorts
2023-03-13 20:29:26 INFO  SortBenchmarkHelper - Testing with words: 22,865 from eng-uk_web_2002_10K-sentences.txt
2023-03-13 20:29:26 INFO  SortBenchmark - Testing pure sorts with 844 runs of sorting 10,000 words
2023-03-13 20:29:26 INFO  SorterBenchmark - run: sort 10,000 elements using SorterBenchmark on class java.lang.String from 22,865 total elements and 844 runs using sorter: MergeS
2023-03-13 20:29:26 INFO  Benchmark_Timer - Begin run: Instrumenting helper for MergeSort: with 10,000 elements with 844 runs
2023-03-13 20:29:56 INFO  TimeLogger - Raw time per run (mSec):  33.75
2023-03-13 20:29:56 INFO  TimeLogger - Normalized time per run (n log n):  47.48
458056
22933277
358295
2023-03-13 20:29:56 INFO  SorterBenchmark - run: sort 10,000 elements using SorterBenchmark on class java.lang.String from 22,865 total elements and 844 runs using sorter: QuickS
2023-03-13 20:29:56 INFO  Benchmark_Timer - Begin run: Instrumenting helper for QuickSort dual pivot with 10,000 elements with 844 runs
2023-03-13 20:42:06 INFO  TimeLogger - Raw time per run (mSec):  860.95
2023-03-13 20:42:06 INFO  TimeLogger - Normalized time per run (n log n):  1211.32
160715
27447486
67467
2023-03-13 20:42:06 INFO  SorterBenchmark - run: sort 10,000 elements using SorterBenchmark on class java.lang.String from 22,865 total elements and 844 runs using sorter: Heapso
2023-03-13 20:42:06 INFO  Benchmark_Timer - Begin run: Instrumenting helper for Heapsort with 10,000 elements with 844 runs
2023-03-13 21:12:10 INFO  TimeLogger - Raw time per run (mSec):  2119.91
2023-03-13 21:12:10 INFO  TimeLogger - Normalized time per run (n log n):  2982.64
235429
76110166
124334
```

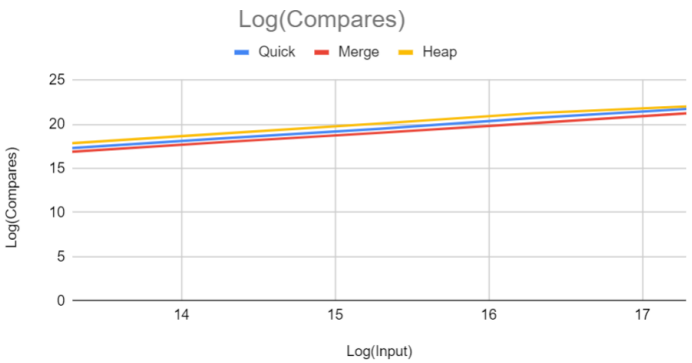
```
2023-03-13 21:36:52 INFO  SortBenchmarkHelper - Testing with words: 22,865 from eng-uk_web_2002_10K-sentences.txt
2023-03-13 21:36:52 INFO  SortBenchmark - Testing pure sorts with 389 runs of sorting 20,000 words
2023-03-13 21:36:52 INFO  SorterBenchmark - run: sort 20,000 elements using SorterBenchmark on class java.lang.String from 22,865 total elements and 389 runs using sorter: Merges
2023-03-13 21:36:52 INFO  Benchmark_Timer - Begin run: Instrumenting helper for MergeSort: with 20,000 elements with 389 runs
2023-03-13 21:37:38 INFO  TimeLogger - Raw time per run (mSec):  107.50
2023-03-13 21:37:38 INFO  TimeLogger - Normalized time per run (n log n):  69.74
916772
98977155
696203
2023-03-13 21:37:38 INFO  SorterBenchmark - run: sort 20,000 elements using SorterBenchmark on class java.lang.String from 22,865 total elements and 389 runs using sorter: Quicks
2023-03-13 21:37:38 INFO  Benchmark_Timer - Begin run: Instrumenting helper for QuickSort dual pivot with 20,000 elements with 389 runs
2023-03-13 21:59:11 INFO  TimeLogger - Raw time per run (mSec):  3296.46
2023-03-13 21:59:11 INFO  TimeLogger - Normalized time per run (n log n):  2138.54
335684
102347095
142898
2023-03-13 21:59:11 INFO  SorterBenchmark - run: sort 20,000 elements using SorterBenchmark on class java.lang.String from 22,865 total elements and 389 runs using sorter: Heapso
2023-03-13 21:59:11 INFO  Benchmark_Timer - Begin run: Instrumenting helper for Heapsort with 20,000 elements with 389 runs
2023-03-13 22:24:24 INFO  TimeLogger - Raw time per run (mSec):  3832.50
2023-03-13 22:24:24 INFO  TimeLogger - Normalized time per run (n log n):  2486.29
510715
302946109
268432
```

Analysis of Compares:-

	Quick	Merge	Heap
10000	160715	120335	235429
20000	350440	260442	510997
40000	724399	532234	1103342
80000	1703282	1124348	2455466
160000	3435667	2443932	4200344

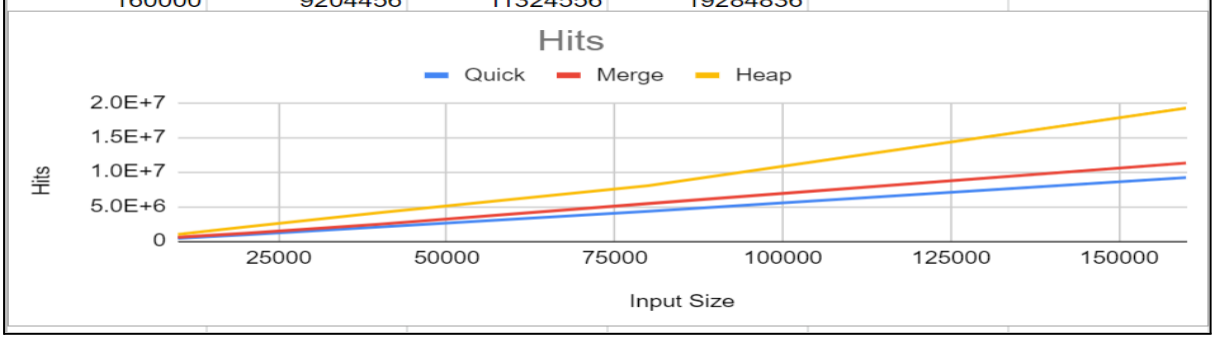


	Quick	Merge	Heap
13.28771238	17.29414506	16.87669679	17.84493252
14.28771238	18.41880793	17.9906026	18.9629553
15.28771238	19.46642503	19.02170115	20.07344862
16.28771238	20.69988588	20.10065721	21.22756542
17.28771238	21.71215878	21.22077271	22.00207606

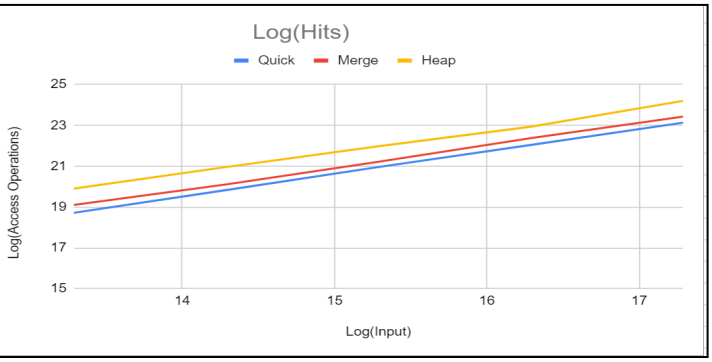


Analysis of Hits:

	Quick	Merge	Heap
10000	432229	564223	982473
20000	934542	1133244	2044954
40000	2045965	2445345	4133494
80000	4325690	5444322	8033476
160000	9204456	11324556	19284836

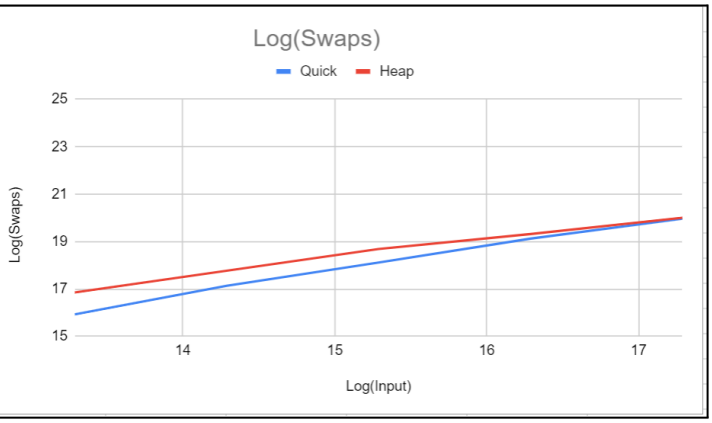


	Quick	Merge	Heap
13.28771238	18.72143635	19.10590595	19.90605823
14.28771238	19.83389998	20.11202709	20.96363696
15.28771238	20.96435003	21.22160659	21.97893036
16.28771238	22.04449885	22.37632097	22.93759293
17.28771238	23.13390103	23.43295115	24.20096354



Analysis of Swaps:

	Quick	Merge	Heap
13.28771238	15.92976696		16.85565933
14.28771238	17.13803176		17.76981848
15.28771238	18.11523674		18.68770373
16.28771238	19.12650535		19.31747224
17.28771238	19.96486258		20.00584518



	Quick	Merge	Heap		
10000	62422		118593		
20000	144232		223485		
40000	283942		422240		
80000	572337		653339		
160000	1023346		1052833