Deep Shah (002766755)

Program Structure & Algorithms

Spring 2023(Sec 03)

Assignment-6

Task:

Task list:

- To determine--for sorting algorithms--what is the best predictor of total execution time: comparisons, swaps/copies, hits (array accesses), or something else.

Relationship Conclusion:

We can conclude that QuickSortDualPivot takes less time to sort the array as compared to Merge Sort and HeapSort. But for larger array sizes merge sort can be as effective as Quick sort.

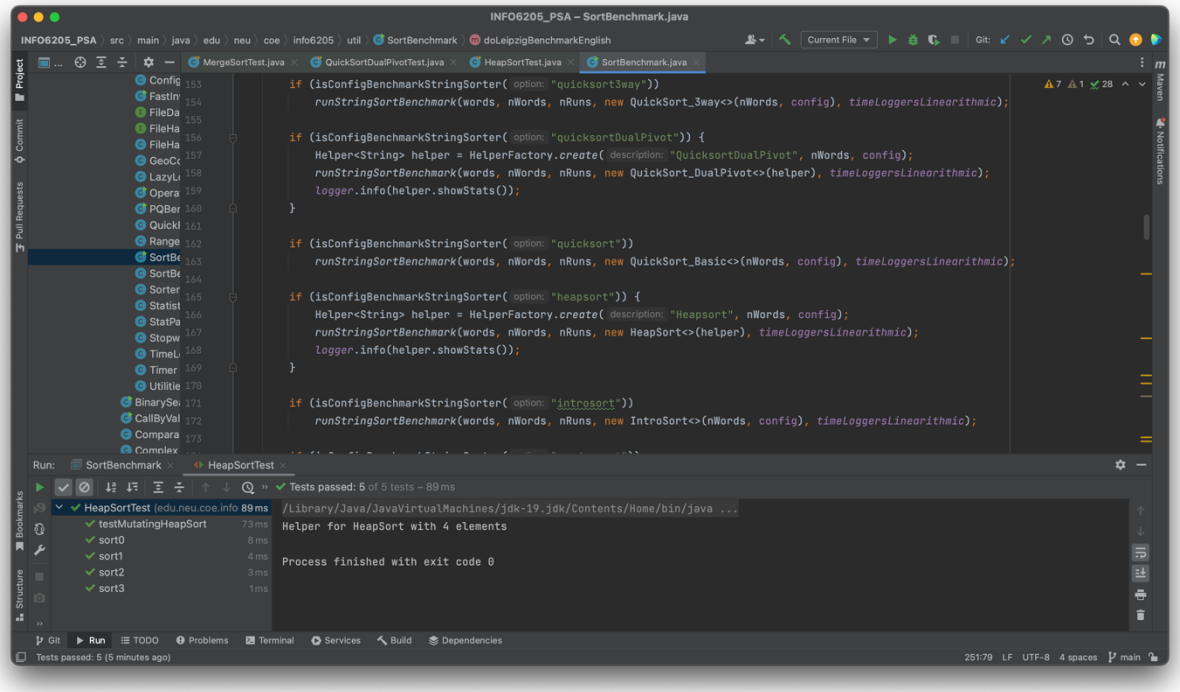
QuickSortDualPivot < Merge Sort < HeapSort

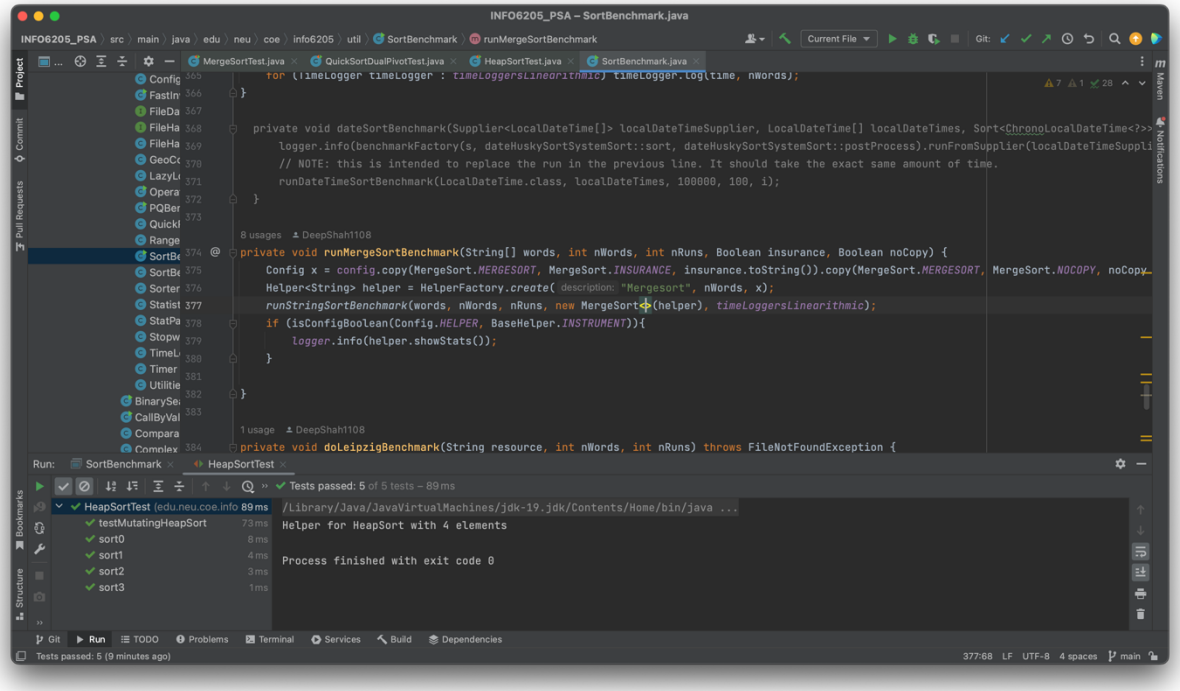
Talking about comparison, swaps, and hits, we can say that as the size of the array increases the no.of comparison, and no. of swapping also increases. Also if the array size increases the time to sort the array also increases with this comparison and swaps.

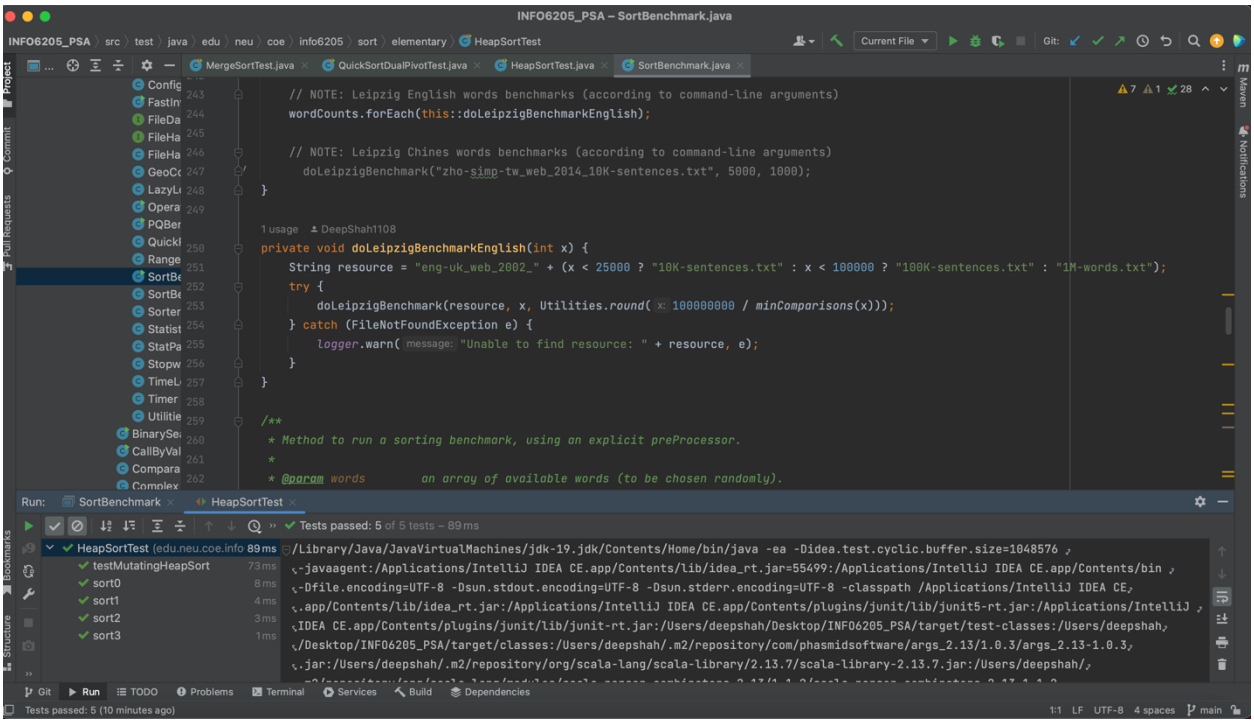
Hence we can say that time taken to sort an array is directly proportional to comparison, swaps, and hit.

Evidence to Support that conclusion:

Running the sorting algorithms(Merge Sort, QuickSortDualPivot, and Heapsort) using the Instrumented Helper and also without the use of Instrumentation.







MergeSort

MergeSort							
Size	**Time**	**Normalized Time**	**Compares**	**Swaps**	**Hits**	**Log(Size)**	**Log(time)**
16000	3.59	2.99	204,955	14,017	434,693	13.96578428	1.843983844
32000	8.57	3.3	441,900	28,022	933,335	14.96578428	3.099295204
64000	19.17	3.44	947,804	56,078	1,994,806	15.96578428	4.260778432
128000	41.03	3.44	2,023,610	112,096	4,245,377	16.96578428	5.358607249
256000	87.16	3.43	4,303,160	224,272	9,003,073	17.96578428	6.445594291

QuickSort(DualPivot)

Size	**Time**	**Normalized Time**	**Compares**	**Swaps**	**Hits**	**Log(Size)**	**Log(time)**
16000	3.34	2.78	265,404	110,666	700,806	13.96578428	1.739848103
32000	6.86	2.64	574,025	242,875	1,530,431	14.96578428	2.778208576
64000	15.54	2.79	1,236,692	516,277	3,273,089	15.96578428	3.957914599
128000	34.92	2.93	2,643,897	1,109,673	7,022,323	16.96578428	5.125981654
256000	75.75	2.98	5,656,293	2,336,965	14,890,396	17.96578428	6.243173983

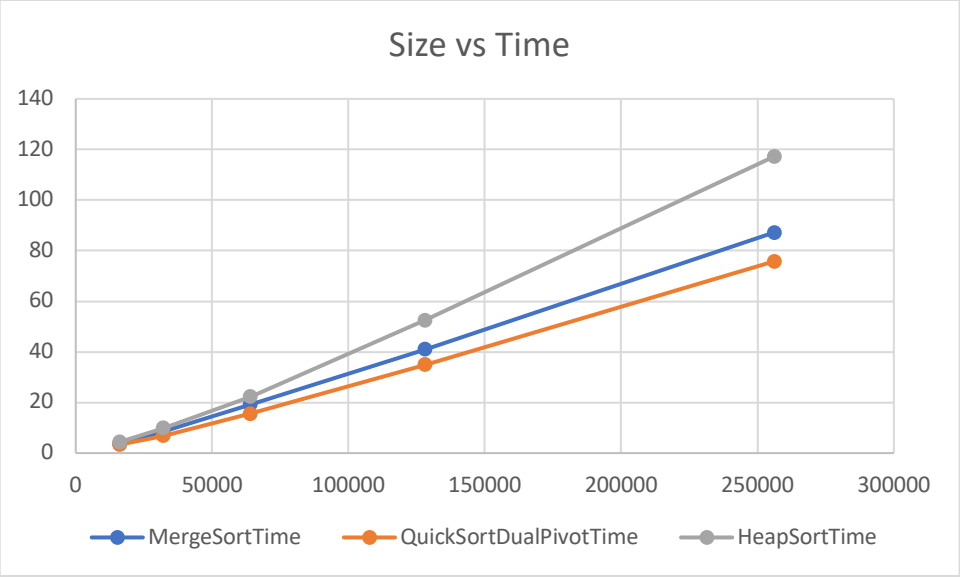
HeapSort

Size	Time	Normalized Time	Compares	Swaps	Hits	Log(size)	Log(time)
16000	4.34	3.61	397,654	209,252	1,632,315	13.96578428	2.117695043
32000	9.8	3.78	859,318	450,511	3,520,681	14.96578428	3.292781749
64000	22.22	3.98	1,846,636	965,028	7,553,384	15.96578428	4.473786912
128000	52.42	4.4	3,949,303	2,058,068	16,130,876	16.96578428	5.712045449
256000	117.16	4.62	8,410,528	4,372,037	34,309,204	17.96578428	6.872336288

Graphical Representation:

1. Size vs Time

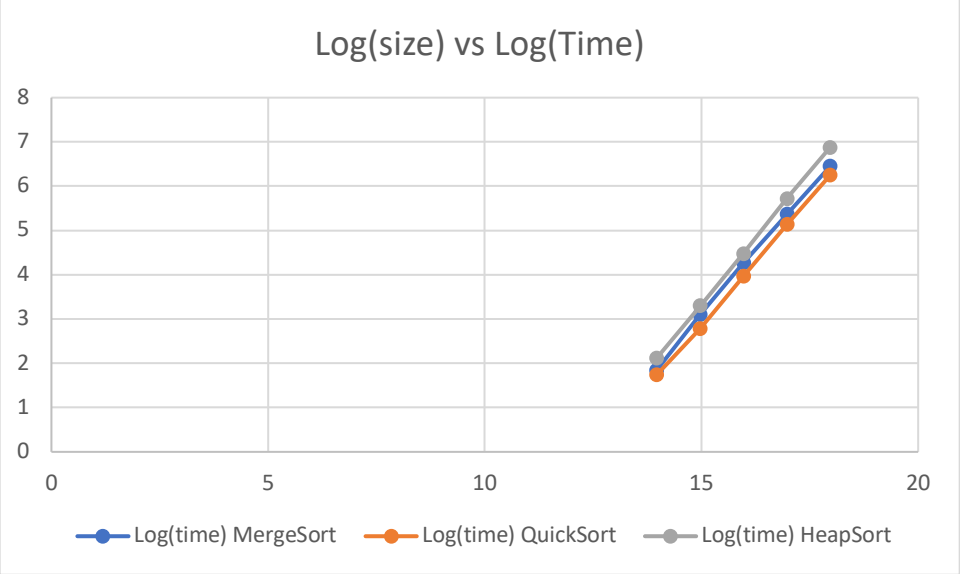
Size	MergeSortTime	QuickSortDualPivotTime	HeapSortTime
16000	3.59	3.34	4.34
32000	8.57	6.86	9.8
64000	19.17	15.54	22.22
128000	41.03	34.92	52.42
256000	87.16	75.75	117.16



2. Log(size) vs Log(Time)

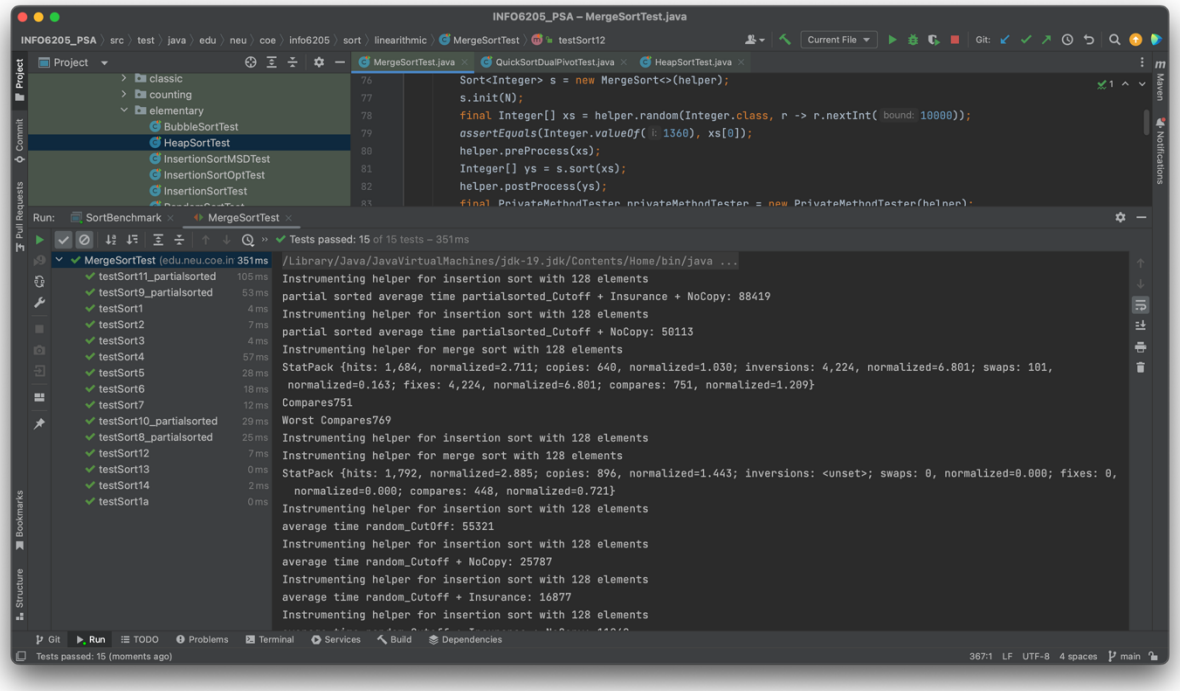
Log(Size)	Log(time) MergeSort	Log(time) QuickSort	Log(time) HeapSort
13.96578428	1.843983844	1.739848103	2.117695043
14.96578428	3.099295204	2.778208576	3.292781749
15.96578428	4.260778432	3.957914599	4.473786912

16.96578428	5.358607249	5.125981654	5.712045449
17.96578428	6.445594291	6.243173983	6.872336288

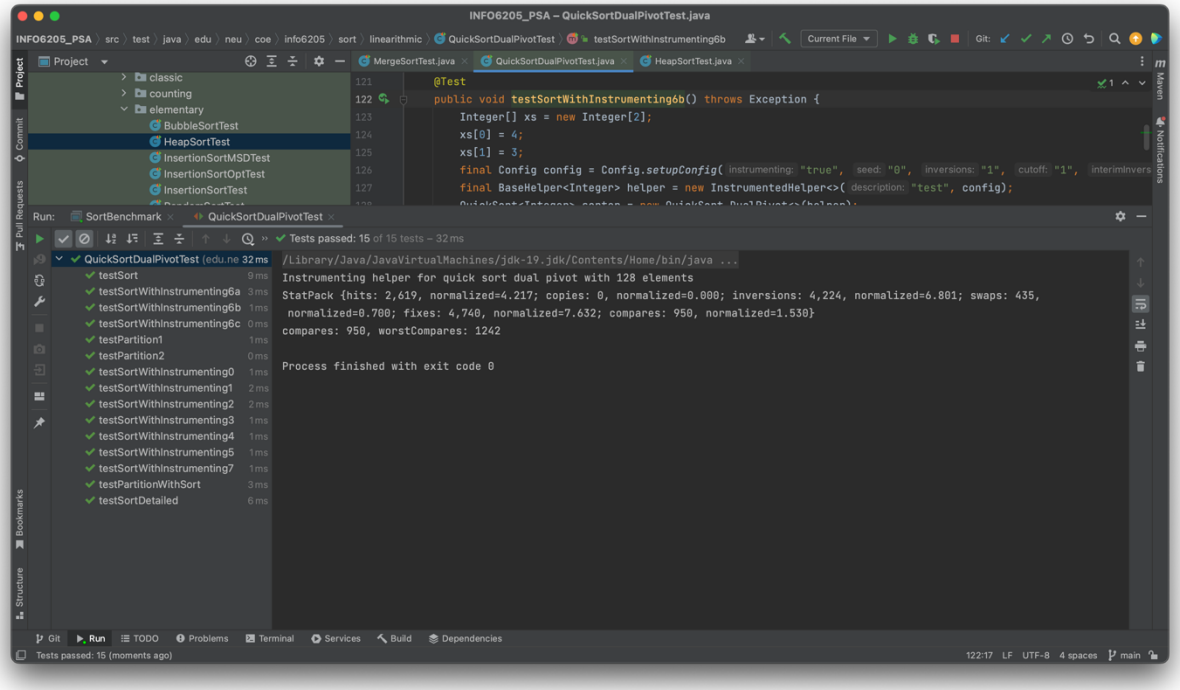


Unit Tests Result:

MergeSort



QuickSort(DualPivot)



HeapSort