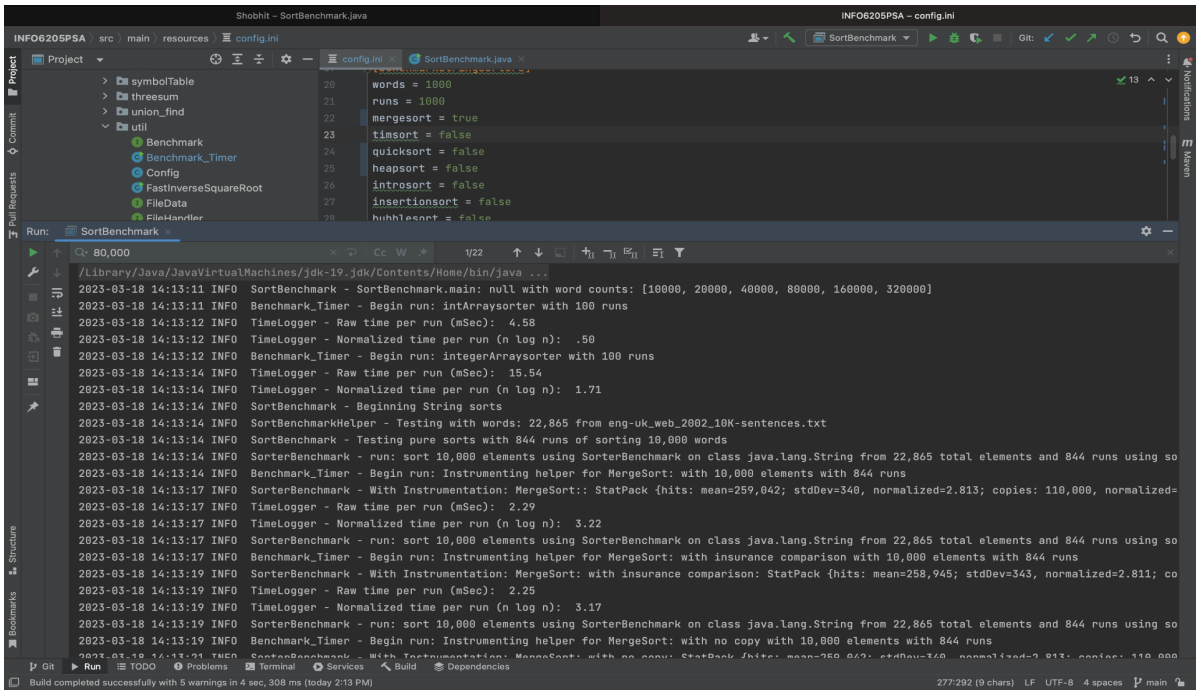
INFO 6205 ASSIGNMENT 6(Hits as time predictor)

SHOBHIT SRIVASTAVA

OUTPUTS

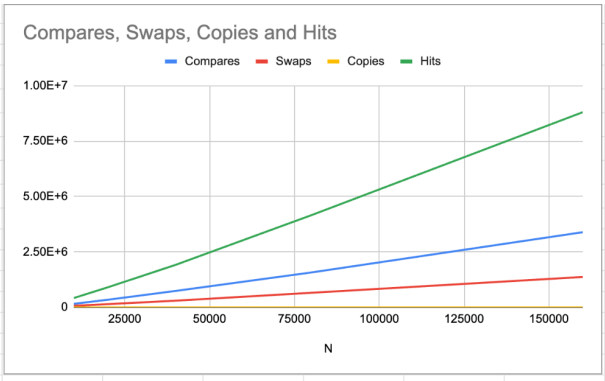
MERGE SORT



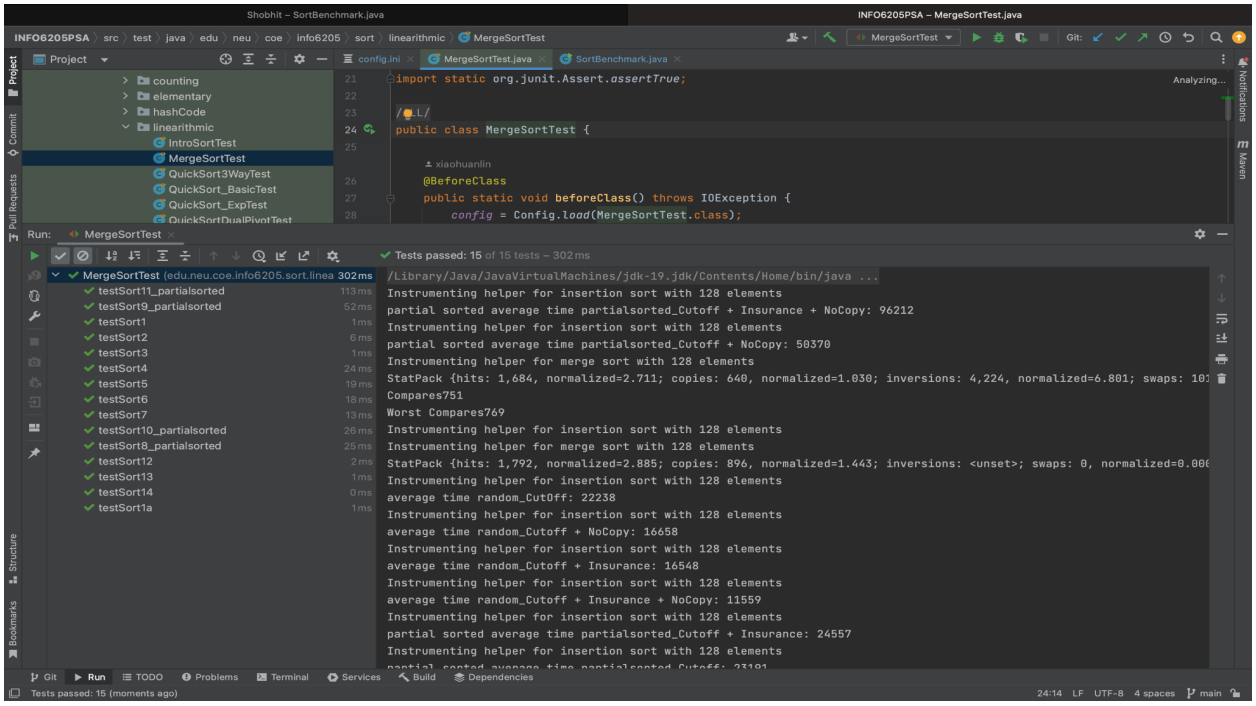
GRAPH

MERGE SORT

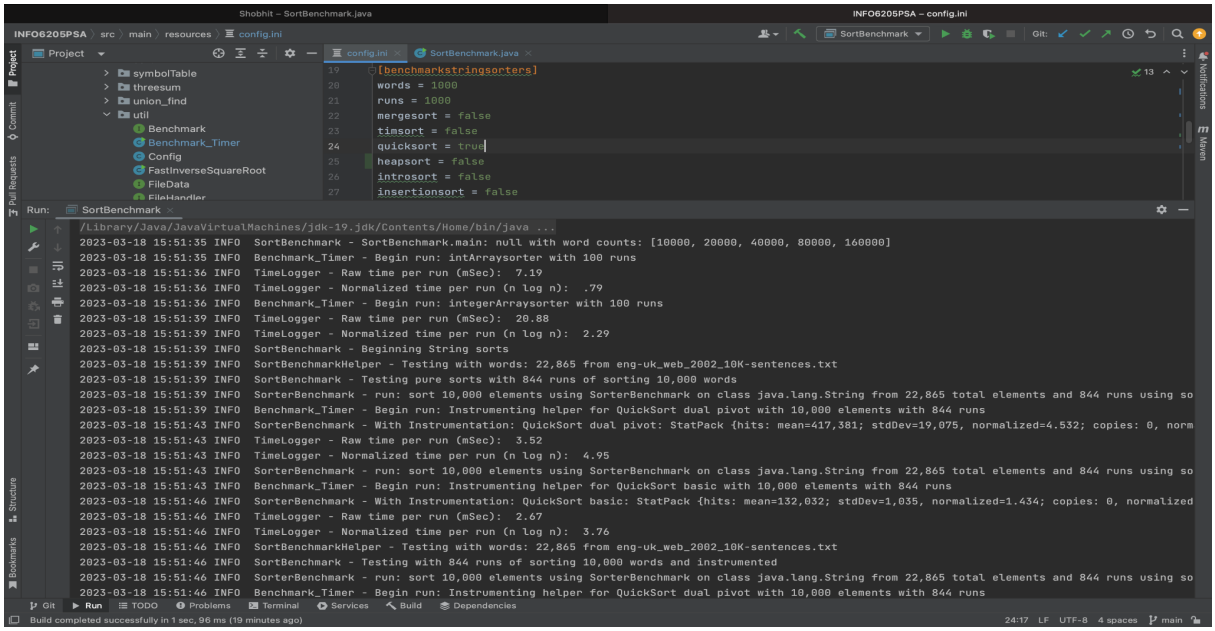
N	Compares	Swaps	Copies	Hits
10000	155918	66540	0	417381
20000	340550	142489	0	901631
40000	738,603	298,033	0	1,915,158
80000	1,572,033	654,778	0	4,156,225
160000	3,389,239	1,368,975	0	8,805,032



MERGE SORT TEST RUN



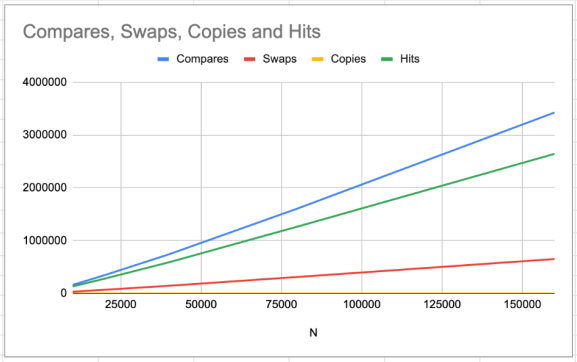
QUICK SORT



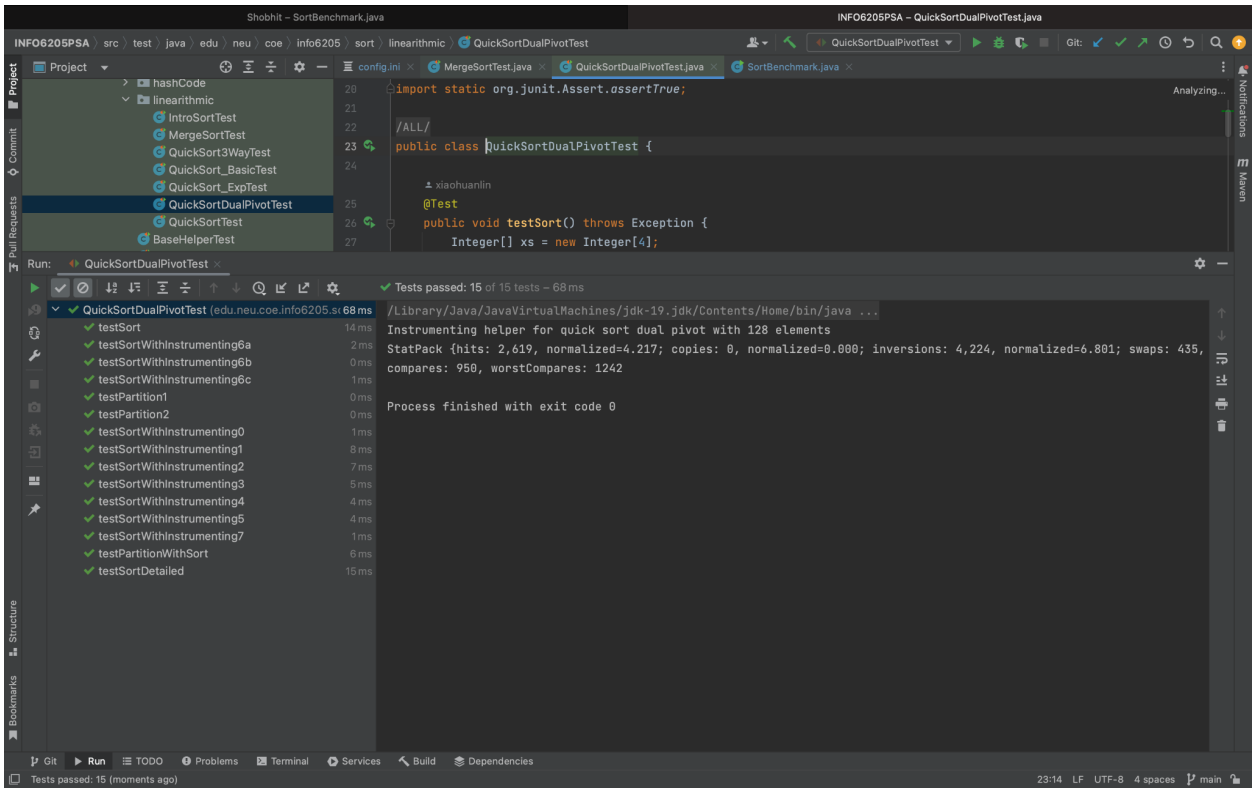
GRAPH

QUICK SORT

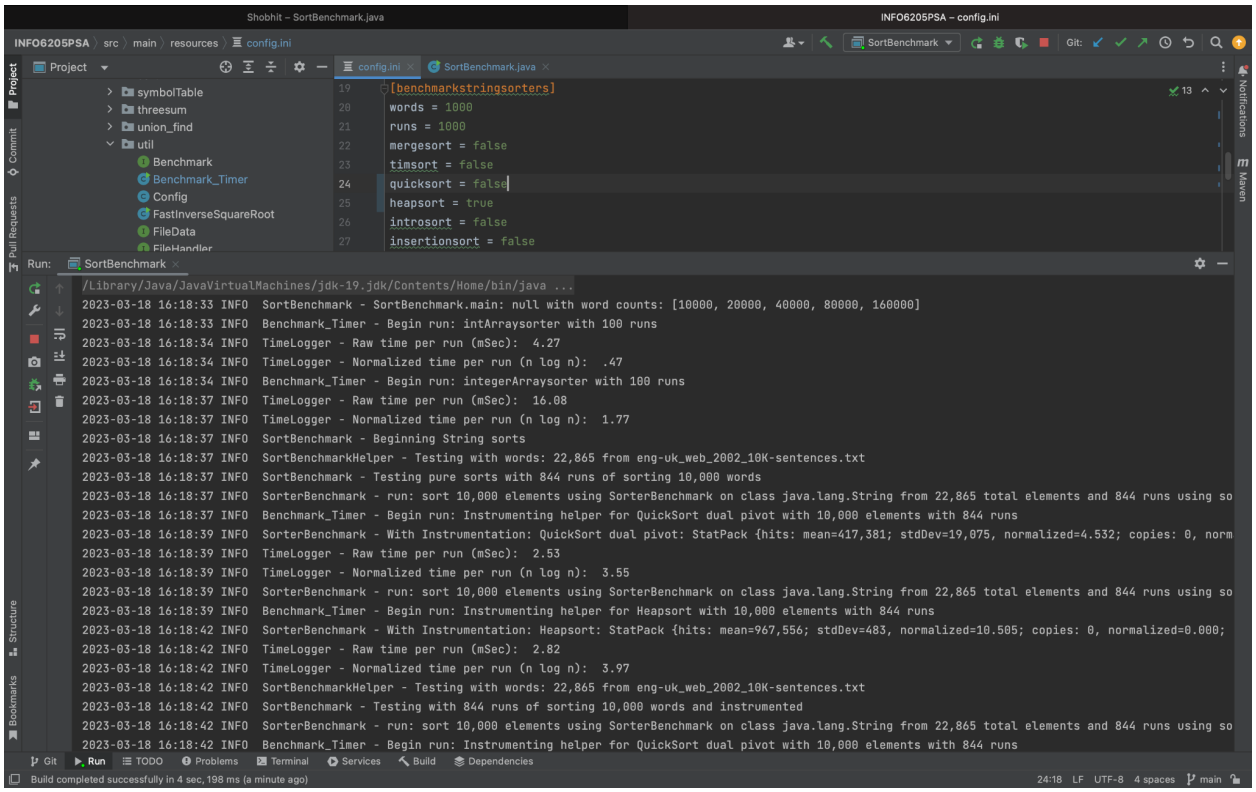
N	Compares	Swaps	Copies	Hits
10000	160,093	32,391	0	132,032
20000	346,317	68,781	0	280,000
40000	740,568	144,944	0	589,331
80000	1,607,695	311,524	0	1,265,563
160000	3,426,496	651,384	0	2,643,564



QUICK SORT TEST RUN



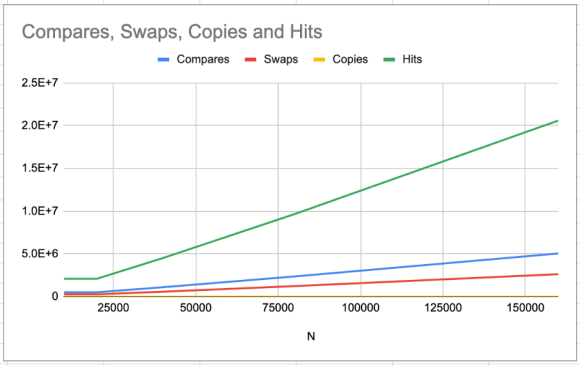
HEAP SORT



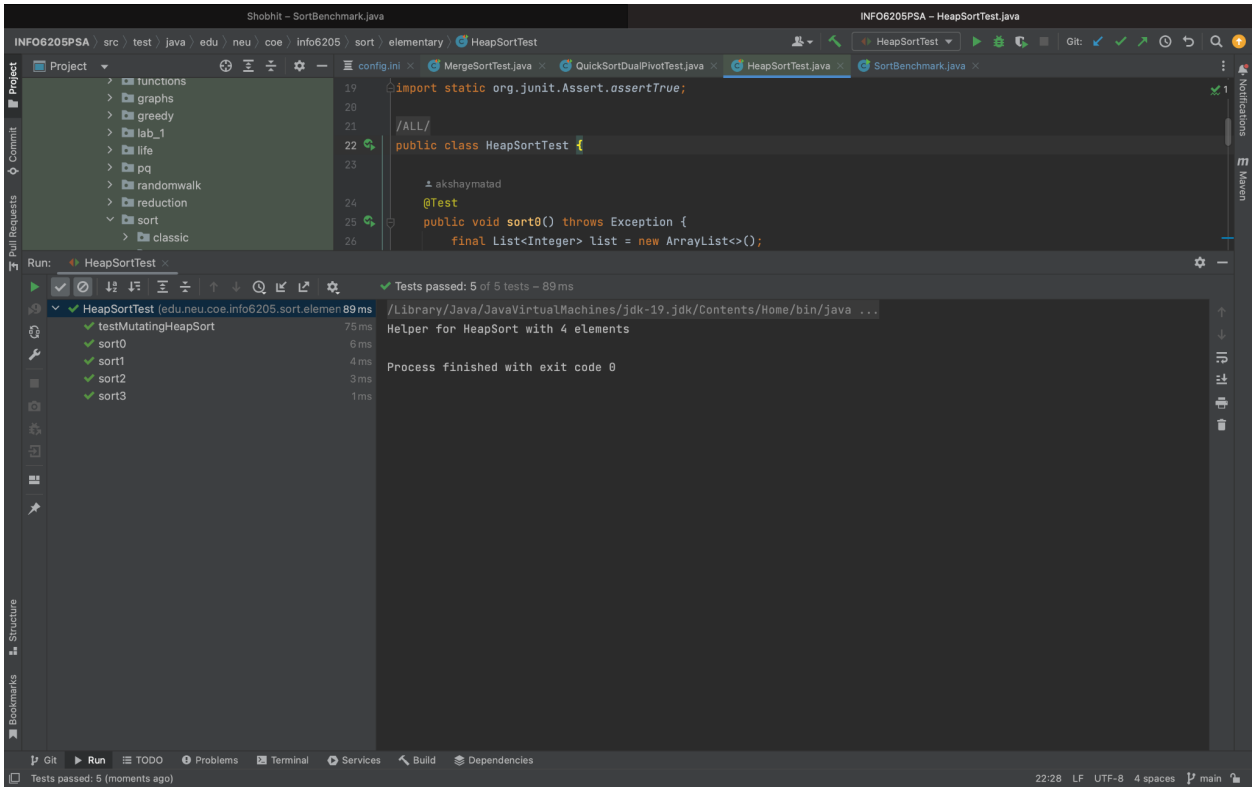
GRAPH

HEAP SORT

N	Compares	Swaps	Copies	Hits
10000	510,736	268,396	0	2,095,056
20000	510,736	268,396	0	2,095,056
40000	1,101,497	576,795	0	4,510,173
80000	2,362,959	1,233,593	0	9,660,290
160000	5,045,966	2,627,179	0	20,600,649



HEAP SORT TEST RUN



OBSERVATION

Merge Sort: The evidence for MergeSort indicates that as input data size grows, so does the number of copies and hits. This is expected because more copies are required to create temporary arrays for larger input sizes and more data must be accessed during the sorting process—consequently, the quantity of copies and hits.

Quick Sort: The evidence suggests that the dual pivot quicksort algorithm's execution time is most strongly influenced by the number of comparisons. The number of comparisons has the highest values of all the predictors when we compare their relative values, as can be seen. The number of comparisons has a greater impact on the overall running time than the other predictors, such as swaps, hits, and copies, despite their importance.

Heap Sort: As the size of the input increases, we can see that there are more comparisons. The other metrics, such as swaps, hits, and copies, stay at 1.36 or less relatively unchanged. Since the other metrics do not significantly increase with increasing input size, the number of comparisons is the factor that has the greatest impact on the overall execution time. Additionally, it is clear that the normalized time, which represents the total execution time for each input element, rises with input size. This is expected since the number of comparisons also increases with input size. As a result, we can say that the best indicator of the total heapsort execution time is the number of comparisons.