**Gilbert Addae-Konadu, Homework 5, CIS671**

\*\*NOTE\*\* The data file must be placed in the root directory(ex. C:/) of your computer when running the program.

**HeapSort**

***Complexity***

O(nlogn)

**InsertionSort**

***Complexity***

best case - O(n)

average/worst case - O(n2)

**QuickSort**

***Complexity***

worst case - O(n2)

average/best case - O(nlogn)

|  |  |  |  |
| --- | --- | --- | --- |
| Elements | HeapSort | InsertionSort | QuickSort |
| 1000 | 0 | 0 | 0 |
| 5000 | 0 | 62 | 0 |
| 10000 | 0 | 203 | 16 |
| 50000 | 32 | 5210 | 31 |
| 100000 | 47 | 20826 | 46 |

Heap sort and quick sort run faster and appear to be more efficient for huge unsorted data we tested with. Insertion sort was the least efficient and took the longest time to sort the data. But for a small quantity of elements they time taken was negligible for all 3 sorting methods. You can tell from the complexities that for the average case, heap sort and quick sort are the same and insertion sort has a larger complexity. This was proven when the program run times were evaluated. Heap sort and quick sort had similar run times and insertion sort a larger run time.