Implementation and comparison the following sorting algorithm:

- **Mergesort**
- **Heapsort**
- **Quicksort (Using median of 3)**
- **Insertion sort**
- **Bubble sort**

By

- Prashant Singh (1001773374)
- Mit Patel (1001740524)

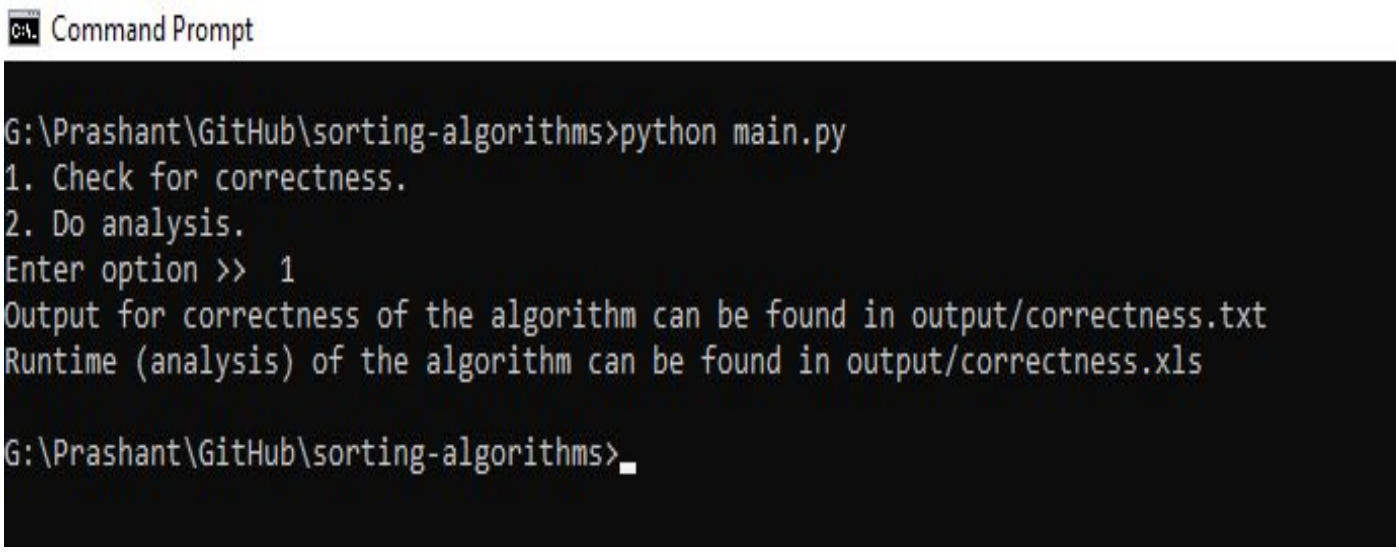
There are two components to our project:

- Correctness
- Analysis

While running the project, you are presented with the above two options and based on the selected option you move forward.

For the implementation of sorting algorithms we have only used Array data-structure throughout for implementation.

1) Correctness: In this section of the code we just check the accuracy of all the sorting algorithms that are being implemented in this project. We are importing an input file which contains 50 predefined list of integers and two files are generated as output for the sorted list of given input integers.



Example:

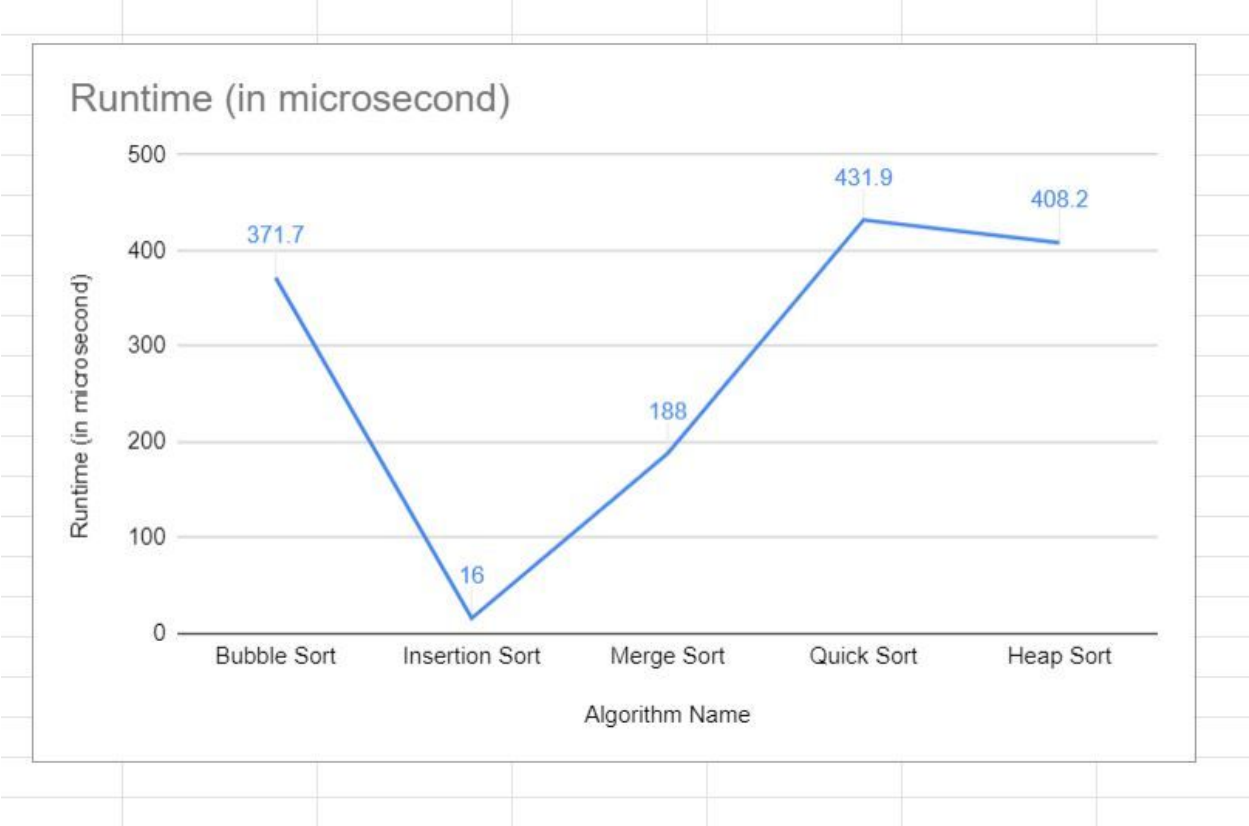
Input: **[14, 5, 55, 61, 56, 60, 49, 95, 86, 46, 78]**

Output of Bubble Sort: [5, 14, 46, 49, 55, 56, 60, 61, 78, 86, 95]
Output of Insertion Sort: [5, 14, 46, 49, 55, 56, 60, 61, 78, 86, 95]
Output of Merge Sort: [5, 14, 46, 49, 55, 56, 60, 61, 78, 86, 95]
Output of Quick Sort: [5, 14, 46, 49, 55, 56, 60, 61, 78, 86, 95]
Output of Heap Sort: [5, 14, 46, 49, 55, 56, 60, 61, 78, 86, 95]

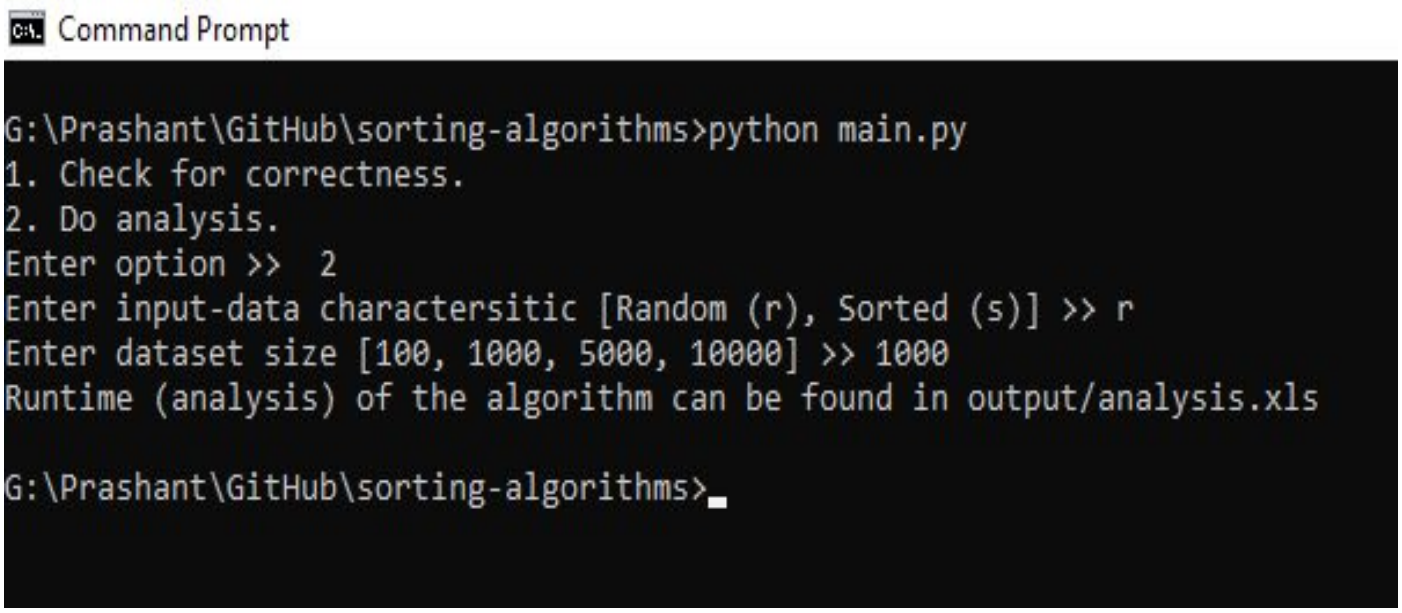
Runtime:

Algorithm Name	**Runtime (in microsecond)**
Bubble Sort	371.7
Insertion Sort	16
Merge Sort	188
Quick Sort	431.9
Heap Sort	408.2

Runtime Graph:



2) Analysis: In this section, the code tries to cover all the edge-cases for the algorithms implemented in the project and demonstrates the change in time-complexity through different input sizes and the characteristic of dataset (sorted and unsorted). The input lengths we considered for this are - 100, 1000, 5000, 10000. The user can select any of the above defined length and the algorithms will consume it, churning out the runtime for the given length of the input in a .xls file.



Number of Inputs / Algorithms	Bubble Sort	Insertion Sort	Merge Sort	Quick Sort	Heap Sort
100	2459.86	93.62	947.04	1127.46	1423.9
1000	108381.76	274.78	6766.68	13371	8370.12
5000	5409584.76	4765.9	68587.78	438074.68	123906.58
10000	18138764.58	2835.94	80115.48	973206.8	130074.16

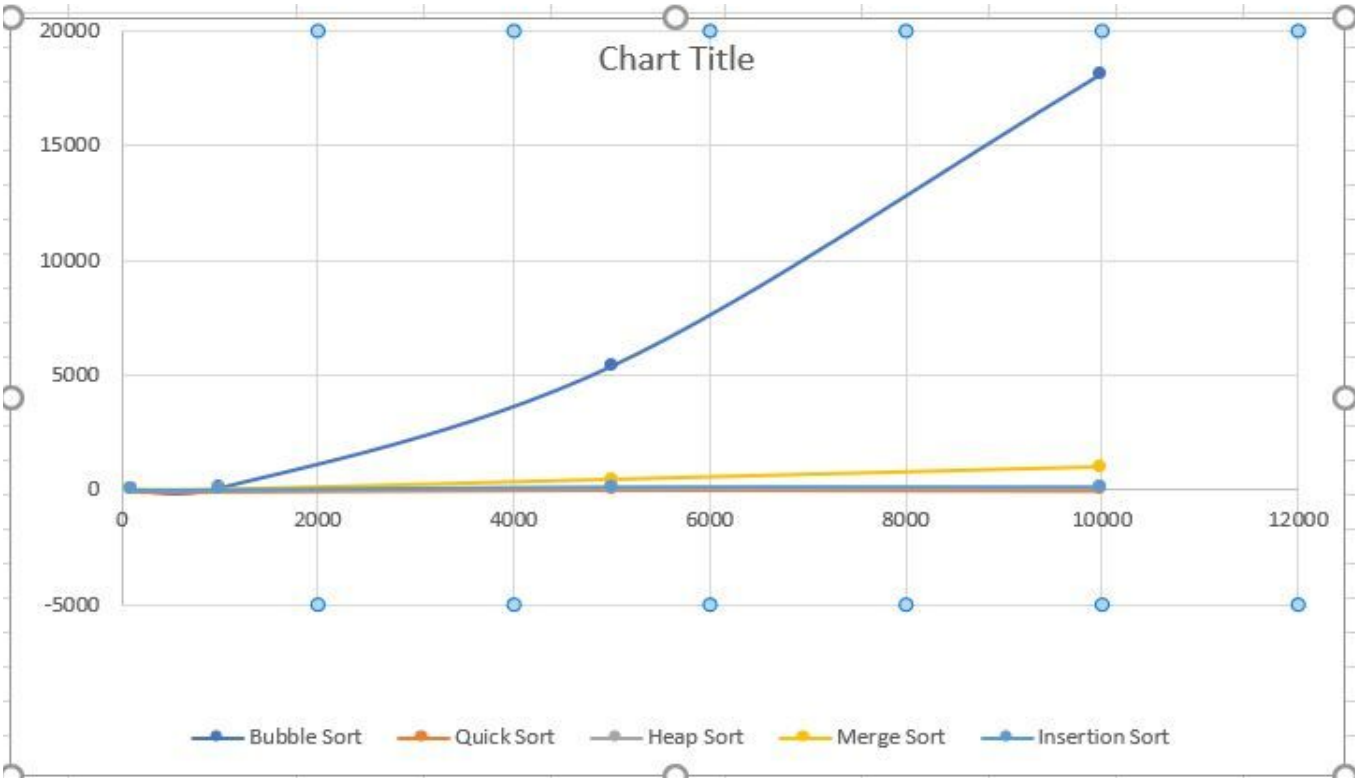


Chart Title
20000
15000
10000
5000
0
-5000
0
2000
4000
6000
8000
10000
12000
Bubble Sort
Quick Sort
Heap Sort
Merge Sort
Insertion Sort