CS 4040/5040

Milestone 1

Author: Viraj Parmar

Date: October 22, 2025

1. **Median Runtime Table (ms)**

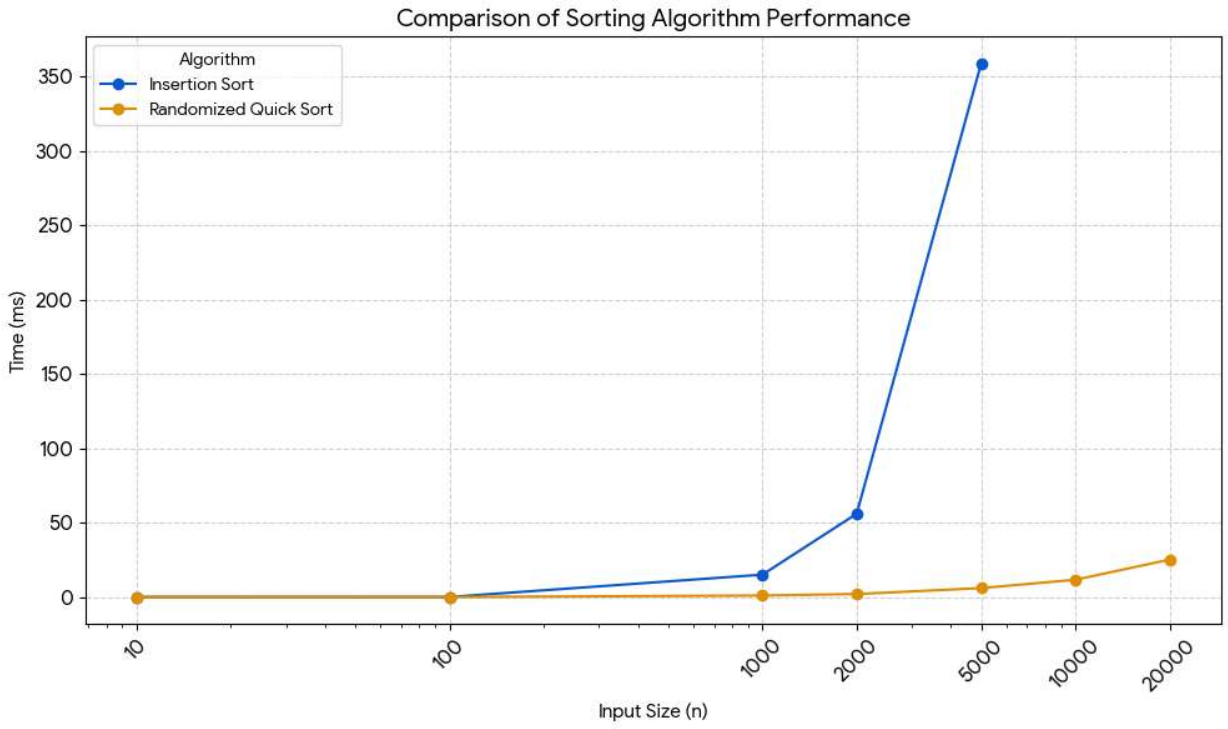
   a. **Insertion Sort**

n	Median Runtime (ms)
10	0.00
100	0.00
1000	15.03
2000	56.03
5000	358.83
10000	Skipped (too slow)
20000	Skipped (too slow)

   b. **Randomized Quick Sort**

n	Median Runtime (ms)
10	0.00
100	0.00
1000	1.00
2000	2.00
5000	6.00
10000	11.60
20000	25.15

2. Runtime Plots:



5. Observations and Analysis:

A. Sorting By Insertion:

- The larger the input size the faster the runtime expands.
- Naturally, the trend follows $O(n^2)$ complexity.
- omitted for very large arrays (n = 10,000 or 20,000) due to the incredibly lengthy runtime.

B. Quick Sorted by Randomization:

- Compared to Insertion Sort the runtime increases more slowly.
- The trend is consistent with theory following O(n log n) complexity.
- up to n = 20,000 efficient even for the massive inputs.

C. Similarity:

- Unpredictable Quick Sort works better with larger arrays than Insertion Sort.
- It is only appropriate for small arrays (n < 5000) to use Insertion Sort.