

## **DESIGN AND ANALYSIS OF ALGORITHM (CSE-5311-008)**

### **Project 1**

**TEAM MEMBERS:** Mrunmai Magar(1002092125)  
Chinmay Hegde(1002029323)

- List any sites/sources referred  
<https://www.scaler.com/topics/merge-sort-in-python/>  
<https://www.scaler.com/topics/insertion-sort-in-python/>  
<https://www.geeksforgeeks.org/quick-sort/> <https://stackoverflow.com/>  
<https://www.geeksforgeeks.org/python-programming-language/>
- Time complexity of algorithms
  1. Insertion Sort:  
Best Case-  $O(n)$   
Average Case-  $O(n^2)$   
Worst Case-  $O(n^2)$
  2. Merge Sort:  
Best Case-  $O(n \cdot \log n)$   
Average Case-  $O(n \cdot \log n)$   
Worst Case-  $O(n \cdot \log n)$
  3. Quick Sort:  
Best Case-  $O(n \cdot \log n)$   
Average Case-  $O(n \cdot \log n)$   
Worst Case-  $O(n^2)$
- Experimental results (time required for algorithms to sort 4 lists of data structures of different lengths).
  1. Insertion Sort  
Array of size 20- 0.000000 seconds  
Array of size 100- 0.000997 seconds  
Array of size 2000- 0.216618 seconds  
Array of size 6000- 2.035130 seconds
  2. Merge Sort  
Array of size 20- 0.000000 seconds  
Array of size 100- 0.000000 seconds  
Array of size 2000- 0.006981 seconds  
Array of size 6000- 0.020944 seconds
  3. Quick Sort  
Array of size 20- 0.000000 seconds

Array of size 100- 0.000000 seconds  
 Array of size 2000- 0.017960 seconds  
 Array of size 6000- 0.055861 seconds

- Compare and contrast the results between the three sorting algorithms and time taken to sort the 4 arrays. Explain anomalies if any:

Arrays	Insertion Sort	Merge Sort	Quick Sort
Size 20	0.000056 seconds	0.000103 seconds	0.000032 seconds
Size 100	0.000589 seconds	0.000558 seconds	0.000142 seconds
Size 2000	0.230511 seconds	0.017761 seconds	0.016035 seconds
Size 6000	1.883815 seconds	0.053697 seconds	0.120081 seconds

Insertion sort is better suited for small arrays, while Merge and Quick sort are better suited for larger arrays.

When we consider an array of size 6000, Quick sort outperforms Insertion sort. Insertion sort takes too long in this case for sorting large arrays.

- Explain any differences between the experimental and theoretical results.

Arrays	Theoretical	Experimental(Average of all the array sort is considered)
Insertion sort	$O(n^2)$	0.5287 seconds
Merge sort	$O(n \log n)$	0.0177 seconds
Quick sort	$O(n \log n)$	0.0340 seconds

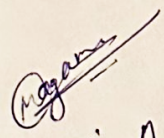
- Participation of each member:  
 Both members of the team worked together on all the coding module and report.

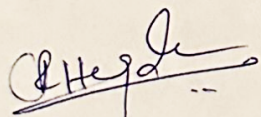
## HONOR CODE

I pledge, on my honor, to uphold UT Arlington's tradition of academic integrity, a tradition that values hard work and honest effort in the pursuit of academic excellence.

I promise that I will submit only work that I personally create or that I contribute to group collaborations, and I will appropriately reference any work from other sources. I will follow the highest standards of integrity and uphold the spirit of the Honor code.

I will not participate in any form of cheating/sharing the questions/solutions.

  
Munmai Nitin Magar  
(1002092125)

  
Chinmay Ramdas Hegde  
(1002029323)

Date: March 5, 2023.