



National Institute of Technology Karnataka, Surathkal

Dileep Reddy Gosu

2420585

CSE

August 30, 2024

Course: M.Tech in Computer Science and Engineering

Course Code & Subject: CS700 & Algorithm and Complexity

Professor: Dr. Vani

Assignment No.: 1

Submission Date: August 30, 2024

Contents

1	Data Generation and Experimental Setup	1
2	Which of the Three Versions of Quicksort Seems to Perform the Best?	1
3	Which of the Six Sorts Seems to Perform the Best (Consider the Best Version of Quicksort)?	3
4	Analyze your data to see if the number of comparisons is correlated with execution time plot (time #comparisons) vs. n and refer to these plots in your answer	5

Introduction

Brief introduction about the assignment.

Question 1

Detailed explanation or answer to question 1.

1 Data Generation and Experimental Setup

1. What kind of machine did you use?
Acer Nitro5
2. What timing mechanism did you use?
c++ Standard Timing
3. How many times did you repeat each experiment?
25 times
4. How did you select inputs?
Randomly with uniform distribution of positive integers in the range of 1 and 1000000
5. Did you use the same inputs for all sorting algorithms?
yes

2 Which of the Three Versions of Quicksort Seems to Perform the Best?

1. Graph the best case running time as a function of input size n for all three versions.

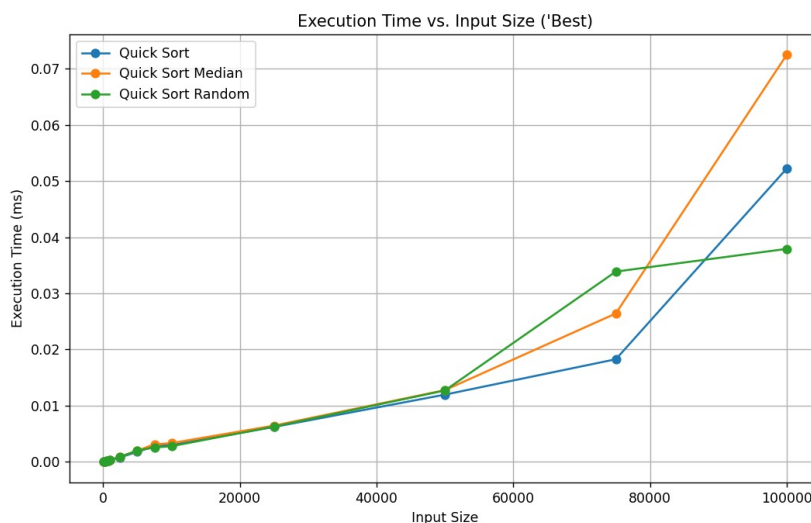


Figure 1: Best Case for all the Sorting Algorithms.

2. Graph the worst case running time as a function of input size n for all three versions.

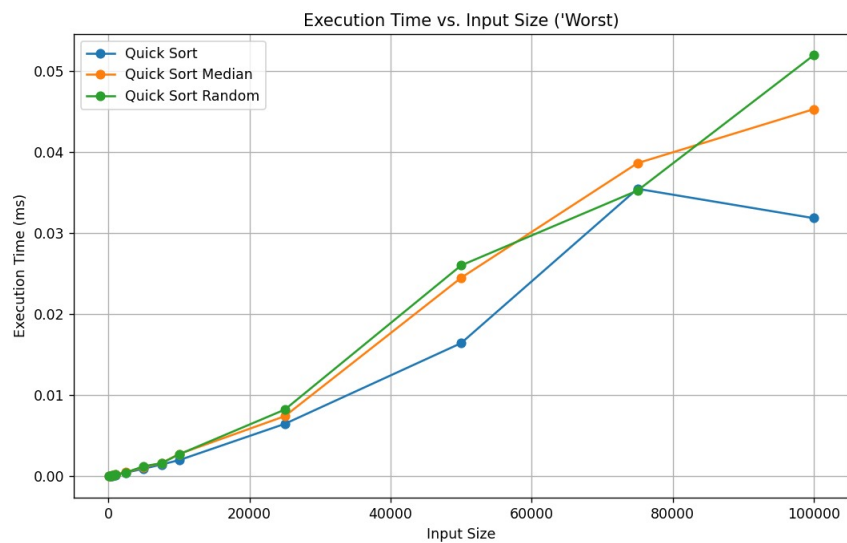


Figure 2: Worst Case for all the Sorting Algorithms.

3. Graph the average case running time as a function of input size n for all three versions.

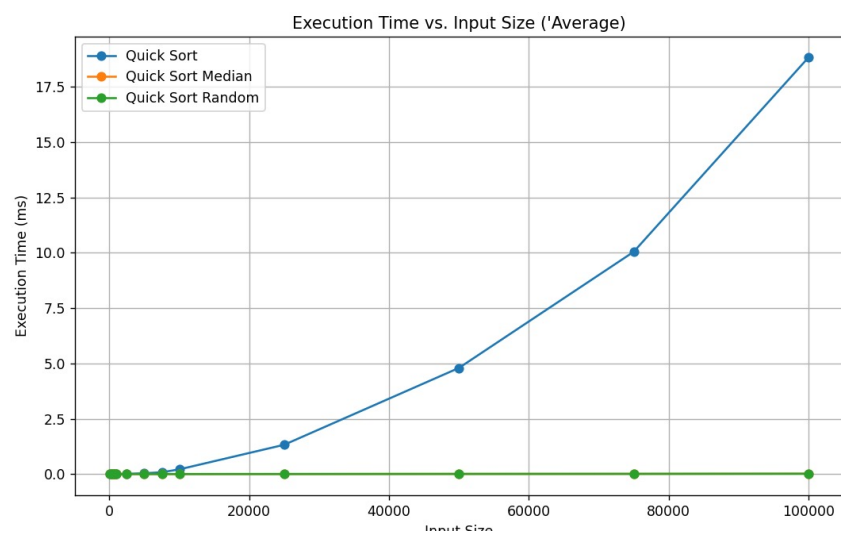


Figure 3: Average Case for all the Sorting Algorithms.

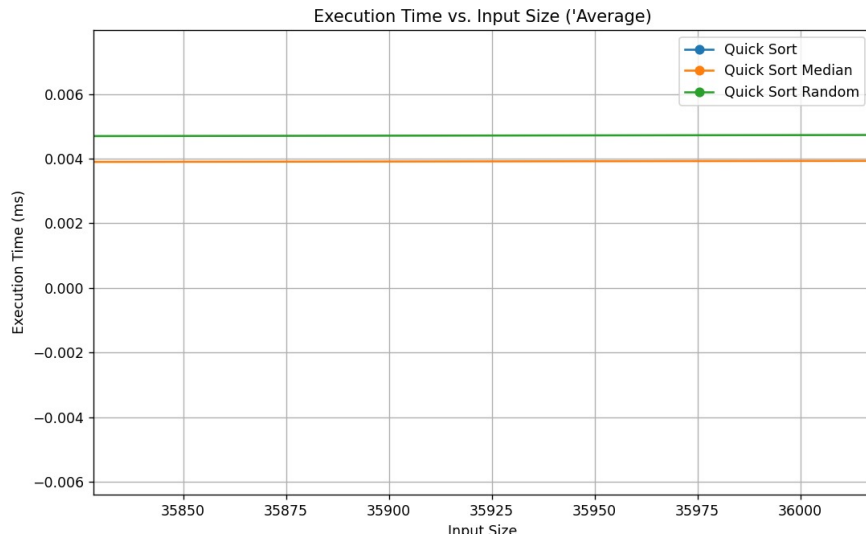


Figure 4: Average Case for all the Sorting Algorithms.Expanded Version

3 Which of the Six Sorts Seems to Perform the Best (Consider the Best Version of Quicksort)?

1. Graph the best case running time as a function of input size n for the six sorts.

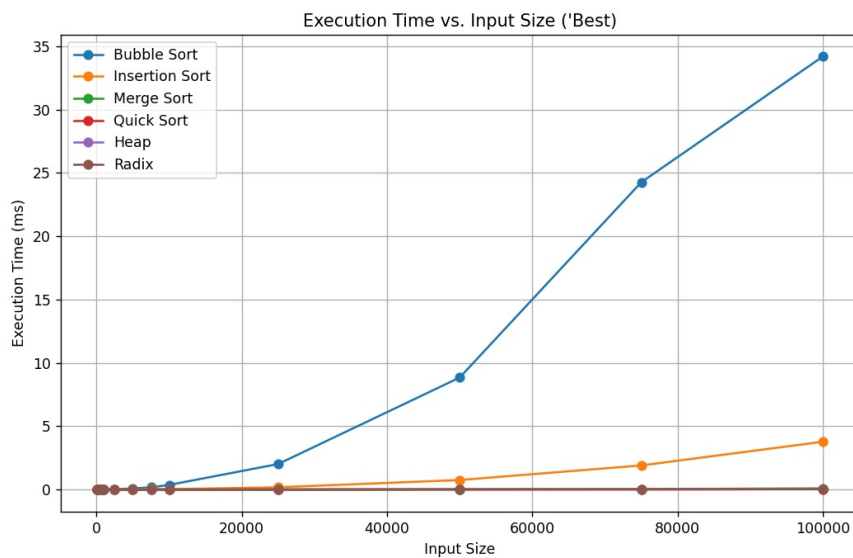


Figure 5: Best Case for all the Sorting Algorithms.

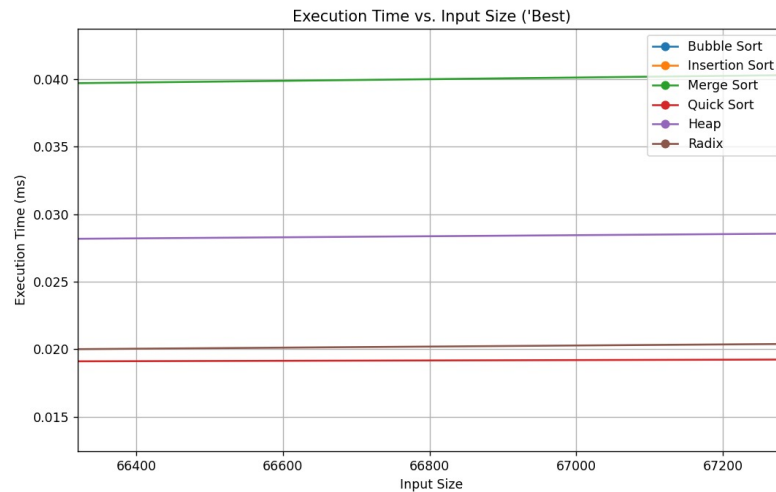


Figure 6: Best Case for all the Sorting Algorithms.Expanded Version

2. Graph the worst case running time as a function of input size n for the six sorts.

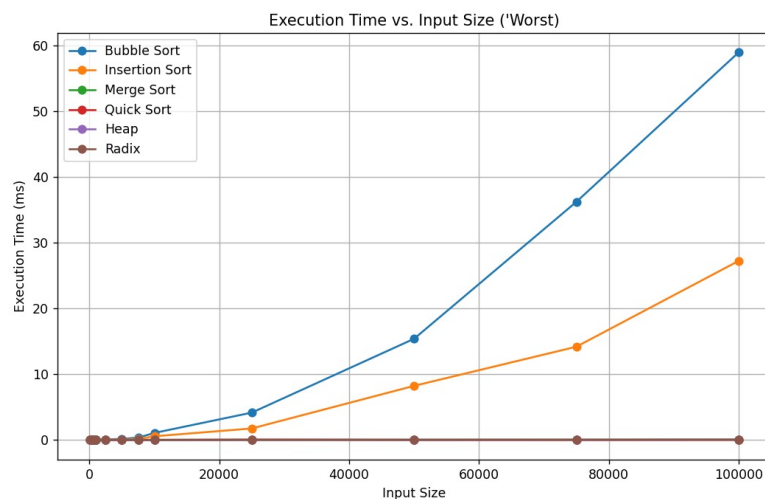


Figure 7: Worst Case for all the Sorting Algorithms.

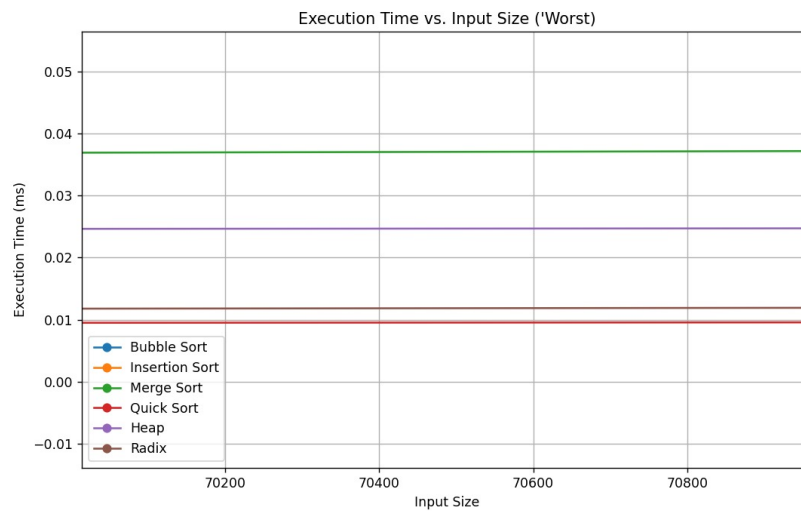


Figure 8: Worst Case for all the Sorting Algorithms. Expanded Version

3. Graph the average case running time as a function of input size n for the six sorts.

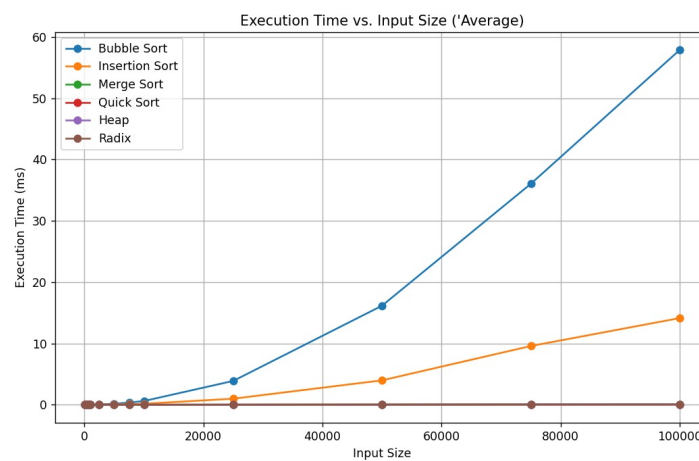


Figure 9: Average Case for all the Sorting Algorithms.

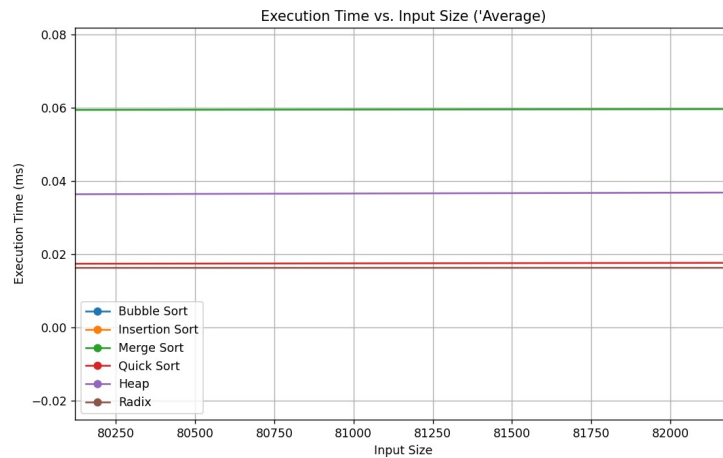


Figure 10: Average Case for all the Sorting Algorithms.Expanded Version

4 Analyze your data to see if the number of comparisons is correlated with execution time plot (time #comparisons) vs. n and refer to these plots in your answer

- **Quicksort:** Showing correlation value nearly 0.99,0.99,0.99 in best and worst case but in average case it is showing 0.99,0.32,0.99 for all 3 types of quicksort variations and for all Algorithms combining it is the fastest algorithm
- **Merge and Heap sort:** Merge sort and heap sort are showing same trend in all the cases
- **Insertion Sort & Bubble sort:** A stronger correlation with n but with higher execution times compared to Merge and Heap Sort
- **Radix Sort:** Shown a linear trend bcox all the number are nearly of same digits

Code: Acces to Github repo