

# Data Structure Project 1

Deadline: Apr 28, 23:59

This project requires students to compare five sorting algorithms, which are “Bubble Sort”, “Insertion Sort”, “Merge Sort”, “Quick Sort”, and “Heap Sort” in the aspect of time complexity, best&worst case scenario.

Requirement:

1. Implement the five sorting algorithms based on the skeleton code provided.
2. Compare the running time of five sorting algorithms, and fill the following table:

t	3	4	5	6	...	14	15	16	17
Bubble									
Insertion									
Merge									
Quick									
Heap									

where each cell in the table denotes the running time (recorded by C++ timer) given the input size (number of elements in the list to be sorted)  $2^t$ . For example, at column “17”, each sorting algorithm should sort the list containing  $2^{17}$  random integers. Note: in order to be fairness to all the sorting algorithms, the input random integer list should be the same.

3. Use “t” as X-axis and running time (value in each cell in above table) as Y-axis, plot all the points and sketch the curve (You may do this by Excel) for each sorting algorithms. Draw all five curves in one X-Y coordinate plane. Compare the five curves and explain the reason.
4. Describe the best/worst case and the corresponding time complexity of each sorting algorithm. You may fill the tables below:

	Best case description	Best case example	Best case time complexity
Bubble			
Insertion			
Merge			
Quick			
Heap			

	Worst case description	Worst case example	Worst case time complexity
Bubble			
Insertion			
Merge			
Quick			
Heap			