Algorithms: Midterm Assignment

Spring 2022

Due: 26 April 2021

Implement the following sorting algorithms in the C programming language and put the described input arrays of different sizes to the algorithms in order to check the answers and the execution times.

- ✓ Algorithm list (six algorithms):
  - ➢ Bubble sort, Insertion sort, Merge sort, and Quicksort.
  - ➢ Radix sort (zero padding if necessary, e.g., 2 → 002 when n is 100).
  - ➢ Bucket sort (bucket number: 50, 250, 500 for the input size of 1,000, 5,000, 10,000, respectively).
- ✓ You should implement the algorithms to sort the input in increasing order.
- ✓ You should run them on a same machine (and not run other programs in the machine) to measure their actual run-time fairly.
- ✓ Input data (consisting of positive integers) – three input data for each algorithm to feed one by one.
  - ➢ Input size n: 1,000, 5,000, and 10,000.
  - ➢ You can simply create the input whose elements are in decreasing order assuming the worst-case.
    - ■ For example, you can create an input of size 1,000 as [1,000, 999, 998, 997, …, 1].
  - ➢ Or, you can create the input whose elements are randomly ordered.
- ✓ You need to report the execution times (in millisecond) of the algorithms for the given sizes.

What you have to submit (through e-class):

- ✓ Implementation codes (screenshots) of the algorithms giving the correct answer.
- ✓ Display (screenshot) of all inputs and outputs of each algorithm.
  - ➢ You can show partial entries for each input and output and make sure that the output is correct.
- ✓ A result table (displayed in console) that gives execution times of the algorithms for different sizes.
  - ➢ x-axis: algorithms, y-axis: input size.
- ✓ All the above requirements should be included in a pdf file (similar to the previous assignment).
- ✓ Note: copying other's codes will simply get zero score with zero tolerance. But, referring to the lecture notes is fine. This is a part of your study on algorithms. You need to implement the algorithms yourself to make your programming and algorithmic skill stronger.

A total of 20 points

- ✓ 3 pts: Bubble sort, Insertion sort, Merge sort, Radix sort.
- ✓ 4 pts: Quicksort, Bucket sort.