COSC 600
Programming Assignment 3
Due date : May 11 (W) by 11:59 pm

Problem 1:

Write two programs to construct a Binary Heap for 5000 random numbers which are between 0 and 50000. These numbers need to be generated by a random function.
One program of constructing a Binary heap is inserting the input elements one at a time, into an initially empty binary heap using insert operation.
The other program of constructing a Binary heap is using the linear-time algorithm to build a heap using same input.

Input: 5000 positive integers generated by a random function ranged between 0 and 50000 and put them into one dimensional array.

Output:

- Calculated the total number of swappings.
- Print out the first 50 elements in the result of each binary heap.
- Print out the execution time for each program.

Problem 2:

Write four sorting programs (insert sorting, selection sorting, bubble sorting, and merge sorting) for 5000 random numbers which are between 0 and 50000. These numbers need to be generated by a random function.

Input: 5000 positive integers generated by a random function ranged between 0 and 50000 and put them into one dimensional array.

Output:

- Print out the first 100 elements in the result of each sorting program.
- Print out the execution time for each program.

Problem 3:

Write a non-traditional counting sorting program for 5000 random which are between 0 and 500. These numbers need to be generated by a random function.

Input: 5000 positive integers generated by a random function ranged between 0 and 500 and put them into one dimensional array.

Output:

- Print out the every 100th elements (50 elements) in the result of this counting sorting program. (it means that print out 100th, 200th, 300th, …. 5000th elements)
- Print out the execution time for this program.

Problem 4:

Write a non-traditional Radix sorting program for 5000 random which are between 0 and 50000. These numbers need to be generated by a random function.

Input: 5000 positive integers generated by a random function ranged between 0 and 50000 and put them into one dimensional array.

Output:

- Print out the first 100 and last 100 elements in ascending order in the result of this sorting program.
- Print out the execution time for this program.