Programming Assignment 2
CSCE 350 : Data Structures and Algorithms
Spring 2023
Due Date - April 11 - Before 10 am

Instructions:

- The solutions should be very clear and should follow the instructions below and the requirements stated for each problem.
- If you refer to any resource to get your solutions, add an acknowledgement with your solutions (details of the source, e.g., book, website, etc.).
- All the codes should be written in c or $c++$ or JAVA for linux and commented appropriately for major steps/functions.
- Code that does not compile will not be graded and get a 0 automatically.
- The codes should be submitted as a single zipped file through Blackboard

Part A:

1. Implement the HeapBottomUp Algorithm using C or C++ or Java. (100 pts)

   Requirements:

   (a) **You are required to implement and submit two separate codes implementing a Max Heap and a Min Heap, where the root node contains the largest and smallest keys, respectively**

   (b) Your code should be able to read an input ASCII file named 'input.txt', where the first line contains the total number of keys, and the second line contains the unsorted keys (integer numbers) separated by blank space

   (c) Your code will produce an output ASCII file named 'output.txt', which contains the resulted heap $H[1, \ldots, n]$ starting from the root, separated by blank space

   (d) Your code should output the execution time for running the algorithm excluding time of input/output.

   (e) A script file or readme file including the instructions to compile and run the code should be submitted together with the codes

Bonus question for Programming assignment 2: Implementing Heapsort Algorithm using C or C++ of Java. (20 pts)

1. Write a C or C++ or Java code to perform Heapsort using your constructed heap.

   Requirements:

   (a) Your code should be able to read an input ASCII file named 'input.txt', where the first line contains the total number of keys, and the second line contains the unsorted keys (integer numbers) separated by blank space

   (b) Your code will produce an output ASCII file named 'output.txt', where the first line contains the resulted heap $H[1, \ldots, n]$ starting from the root, separated by blank space, and the second line contains the heapsort result starting from the largest number

   (c) A script file or readme file including the instructions to compile and run the code should be submitted together with the codes