CSci 242: Algorithms and Data Structures **Spring, 2022**

Instructor: Dr. M. E. Kim Date: February 22nd, 2022

Due: by the end of day, March 1st (Tue.), 2022.

**Home Assignment 4: 100 points**

Read the submission instruction and comply with it; otherwise, it won’t be graded.

Q1. [40] **Maximum/Minimum Heap and Heap Sort**

In the array A[1..10] that stores the keys [2, 5, 16, 4, 10, 23, 39, 18, 26, 15],

1. [10] ***Construct*** a ***max-heap*** by *in-place heap construction* in the array A.

During the construction of heap, show each step of heap construction.

Draw the final max-heap and show its content in the array A. – Do not use ‘BottomUpHeap’ construction.

![Diagram

Description automatically generated]()

1. [10] (A) Draw the max-heap after ***removing the Maximum key*** from the final max-heap of 1), and (B) show its content in the array A.

![Diagram, schematic

Description automatically generated]()![Chart, scatter chart

Description automatically generated]()

1. [10] ***Construct*** a ***min-heap*** in the original array A. Draw the final min-heap and show its content in the array A.

Background pattern

Description automatically generated with medium confidence

1. [10] From the min-heap in 3), ***sort*** the array A in the ***descending*** order.
2. Show the content of array A***after sorting five smallest keys***, and
   1. Unsorted array A after min heap

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 4 | 16 | 5 | 10 | 23 | 39 | 18 | 26 | 15 |

* 1. Sorted array

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 15 | 10 | 5 | 4 | 2 |

1. Draw the final-heap after sorting and show the content of the final array A.

A picture containing graphical user interface

Description automatically generatedBackground pattern

Description automatically generated

1. [10, optional] Suppose that a min-heap is implemented as the *list-based heap*, not in an array.

Write the algorithms of heap insert(*k*) and removeMin in pseudo codes.

Q2. [10] **Heap Sort**

A minimum heap is used to sort an array A of length *n* in descending order.

What is the running time of Heap-Sort algorithm on an array A that is *already sorted* in descending order? Explain your answer.

The running time of a heap-sort algorithm in descending order would be Big-Theta (n log n) time because every iteration of the algorithm a heapify algorithm is ran to ensure that each parent is less than its children for n elements in the heap.

Q3. [10] Insertion Sort

In the array A[1 .. 10] that stores the keys [22, 15, 36, 44, 10, 3, 9, 13, 29, 25], sort A in the descending order by Insertion sort. Show/(print) the changes of the array at *each step*.

Graphical user interface

Description automatically generated with medium confidence

Q4. [10] Selection Sort

In the array A[1 .. 10] that stores the keys [22, 15, 36, 44, 10, 3, 9, 13, 29, 25], sort A in the descending order by Selection sort. Show/(print) the changes of the array at *each step*.

Table

Description automatically generated

Q5. [30] **Implementation** of Q1**.**

Implement the operations/algorithms in Q1.(1) – (4), Q3, and Q4 using Python and print the outputs. Insert the image(s) of the outputs.

Q1(1) Text

Description automatically generated

Q1(2) Text

Description automatically generated with low confidence

Q1(3) A screen with numbers and letters on it

Description automatically generated with low confidence