

## LAB-VI

Date: **Sep 05, 2024.**

You need to upload your solutions of Q1-Q3 (Q4 is optional) to canvas portal before 05:35pm on Sep 05, 2024.

In this Lab, in Q1-Q3, we work with the array to represent heaps (for details refer to lecture notes).

1. Let  $A$  be an array having  $n$  integers. Write a program to convert the array  $A$  into a MAX HEAP.
2. Given an array  $A$  having  $n$  integers, write a program to sort  $A$  using HEAP SORT.
3. Write a program to implement a priority queue  $Q$  using MAX-HEAPS. Your priority queue should support the following operations.
  - $\text{INSERT}(Q, x)$ : inserts the element  $x$  into the queue  $Q$ .
  - $\text{MAXIMUM}(Q)$ : returns the element of  $Q$  with the largest key.
  - $\text{EXTRACT-MAX}(Q)$ : removes and returns the element of  $Q$  with the largest key.
  - $\text{INCREASE-KEY}(Q, x, k)$ : increases the value  $x$ 's key by  $k$ .

All the above operations should run in  $O(\log n)$  time, where  $n$  is number of elements in  $Q$ .

4. Given an array of distinct integers your goal is to convert it into a max-heap. You are only allowed to do 'swap' operations on array. The swap operation exchanges elements  $a_i$  and  $a_j$  for some  $i$  and  $j$ .

Your input format is as follows: The first line contains single integer  $n$  (size of the array). The next line contains  $n$  integers separated by a space. Your output should be as follows. The first line of the output should be a single integer  $m$  (the minimum number of swaps needed). The next  $m$  lines should contain the swap operations used to convert the array into a max-heap. Each swap is described by a pair of integers  $i$  and  $j$  (indices of the elements to be swapped). After applying all the swaps in the specified order the input array must become a max-heap.

You have to read input from a file named 'heap.txt'

Sample Input 1 :

5  
1 2 3 4 5

Output:

2  
0 4  
1 3

Description: After swapping element 1 in position 0 and 5 in position 4 the array becomes 5 2 3 4 1. After swapping element 2 in position 1 and 4 in position 3 the array becomes 5 4 3 2 1 (which is a max-heap).