Algorithms -1
Tutorial 4
August 24, 2018

1. Show the min-heap that will be formed after inserting the elements 23, 1, 34, 4, 56, 77, 12, 33, 2, 12, 54, 2, 1, 34 one by one in an empty heap? Would the heap be the same if the elements are given at one time in an array and you use the Build-Heap procedure to construct a heap out of it? .

2. Given two arrays A1 and A2 (each of size n) of integers, design an O(nlgn) time algorithm for finding whether there exists a pair of elements, one from A1 and one from A2, such that their sum is some given integer x.

3. Consider an array A of n integers such that $A[0] \leq A[1] \geq A[2] \leq A[3] \geq A[4] \leq A[5] \geq A[6] \leq ...$ Can you sort it in O(n) time?

4. Given two heaps each with n elements, how can you merge them to create a single heap of 2n elements in O(n) time?

5. Suppose that you are given n integers in a min-heap (already formed), and an integer m. Design an algorithm to find all elements in the heap with value < m in O(k) time, where k is the number of elements in the hep that have value < m.

6. You are given k sorted arrays $A_1, A_2,...,A_k$. Design an algorithm to merge them in O(nlgk) time algorithm to merge the k sorted arrays into a single sorted array, where n is the total number of elements in all the arrays.

7. You are given an integer n. You are also given at most n (x_i, v_i) pairs such that all x_i are distinct integers and $1 \leq x_i \leq n$. You need to design an ADT which supports the following operations:
    a. getMax(): Get the maximum value v among all the values present. Time : O(1)
    b. decreaseKey(x, v): Decrease the value of key x to v. You can assume v is always less than the current value of x. Time : O(log n)

    You are allowed to pre-process the data before the operations in O(n) time.

8. Given an array of n integers, design an O(nlgn) algorithm to find the majority element (element that occurs more than n/2 time) in the array. Try two variations (i) if you are allowed to use sorting, and (ii) if you are not allowed to use sorting. Can you do it in O(n) time?

9. Suppose that a stream of integers are coming one by one, and you have to find the median of the elements after inserting each integer in O(lgn) time. Design an algorithm to do this. You are not allowed to use any balanced BST.