

Practical Assignment

Data Structures and Algorithms Lab (CS 253)

October 12, 2022

Lab Assignment VII

Note: For problems 3-6, try to solve using binary heap or AVL trees instead of attempting to solve the problem in a brute-force approach

1. Write a C++ program that takes the level order traversal of a binary tree as input, and identifies if it satisfies the min-heap property or not.
2. Write a C++ program to implement a priority queue using Binary Heap (Note: Do not use STL)
3. Given a sorted integer array “arr” containing the element “1” and prime numbers, where all the integers of arr are unique. You are also given an integer k. For every i and j where $0 \leq i < j < \text{arr.length}$, we need to consider the fraction $\text{arr}[i] / \text{arr}[j]$.

Write a program that returns the kth smallest fraction considered. Return your answer as an array of integers of size 2, where $\text{answer}[0] == \text{arr}[i]$ and $\text{answer}[1] == \text{arr}[j]$.

Lab Assignment VII

4. Given a non-empty string s and an integer k , write a program to rearrange the string such that the same characters are at least distance k from each other. All input strings are given in lowercase letters. If it is not possible to rearrange the string, return an empty string `" "`.

For eg.

Input: $s = \text{"aabbcc"}, k = 3$

Output: `"abcabc"`

Explanation:

The same letters are at least distance 3 from each other.

Input: $s = \text{"aaabc"}, k = 3$

Output: `""`

Explanation:

It is not possible to rearrange the string.

Input: $s = \text{"aaadbcc"}, k = 2$

Output: `"bacabacd"`

Explanation:

Another possible answer is: `"abcabcda"`

The same letters are at least distance 2 from each other.

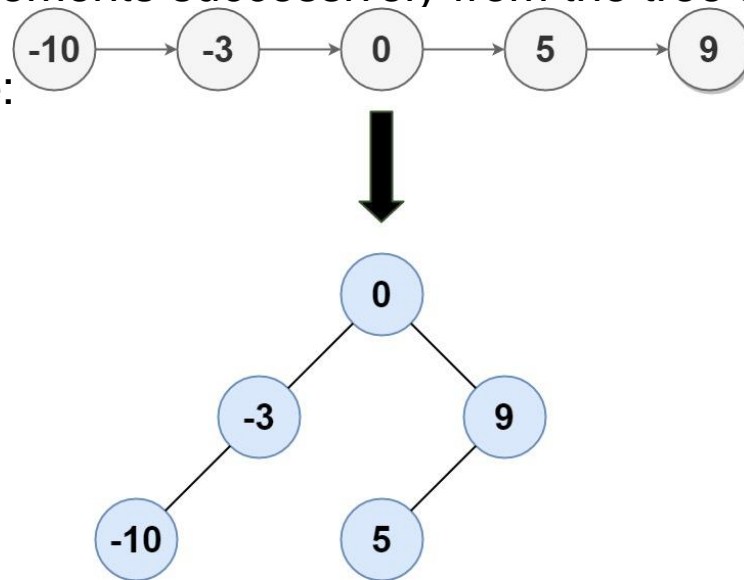
Lab Assignment VII

5. Given the head of a singly linked list where elements are sorted in ascending order, write a program to convert it to a height-balanced BST.

For this problem, a height-balanced binary tree is defined as a binary tree in which the depth of the two subtrees of every node never differs by more than 1. And, then delete two elements successively from the tree and balance it again after deletions,

For Example:

Example 1:



Input: head = [-10,-3,0,5,9]

Output: [0,-3,9,-10,null,5]

Explanation: One possible answer is [0,-3,9,-10,null,5], which represents the shown height-balanced BST.

Lab Assignment VII

6. Given that integers are read from a data stream. Write a program to find the median of elements read so far in an efficient way. For simplicity assume, there are no duplicates.

For example, let us consider the stream 5, 15, 1, 3 ...

The output should be:

After reading 1st element of stream - 5 -> median - 5

After reading 2nd element of stream - 5, 15 -> median - 10

After reading 3rd element of stream - 5, 15, 1 -> median - 5

After reading 4th element of stream - 5, 15, 1, 3 -> median - 4,
so on...

To make it clear, when the input size is odd, we take the middle element of sorted data. If the input size is even, we pick the average of the middle two elements in the sorted stream.

Note that output is the effective median of integers read from the stream so far.