Week 12- Day 3 : Coding Challenge

(Maximum marks 15)

Q-1) write a program to take input a Binary tree and tell if that binary tree is balanced or not?

https://leetcode.com/problems/balanced-binary-tree/

(5 marks)

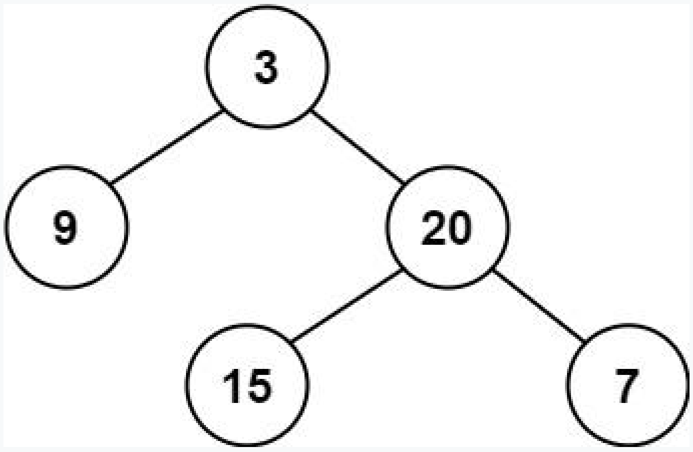
(Easy)

Given a binary tree, determine if it is height-balanced.

For this problem, a height-balanced binary tree is defined as:

a binary tree in which the left and right subtrees of *every* node differ in height by no more than 1.

Example 1:



Input: root = [3,9,20,null,null,15,7]

Output: true

Q-2)Write steps in heapify/percolate down method, and write time complexity and space complexity analysis.(5 marks)

(Super Easy)

Q - 3) Merge Two Binary Trees (5 marks)
https://leetcode.com/problems/merge-two-binary-trees/

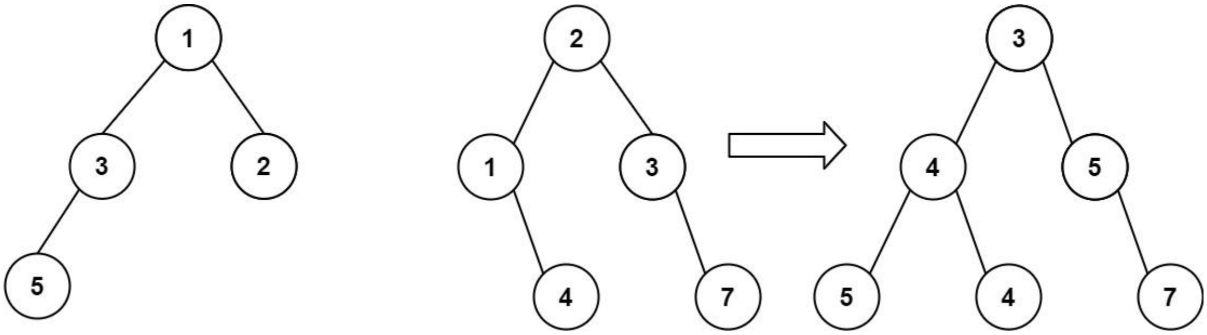
(Easy)
You are given two binary trees root1 and root2.

Imagine that when you put one of them to cover the other, some nodes of the two trees are overlapped while the others are not. You need to merge the two trees into a new binary tree. The merge rule is that if two nodes overlap, then sum node values up as the new value of the merged node. Otherwise, the NOT null node will be used as the node of the new tree.

Return *the merged tree*.

Note: The merging process must start from the root nodes of both trees.

Example 1:



Input: root1 = [1,3,2,5], root2 = [2,1,3,null,4,null,7]
Output: [3,4,5,5,4,null,7]

Example 2:

```
Input: root1 = [1], root2 = [1,2]
Output: [2,2]
```

Marks distribution:
Question 1 and 2 and 3 carry 5 marks each.