

Day 5 - Arsh Challenge

Maximum depth of binary tree

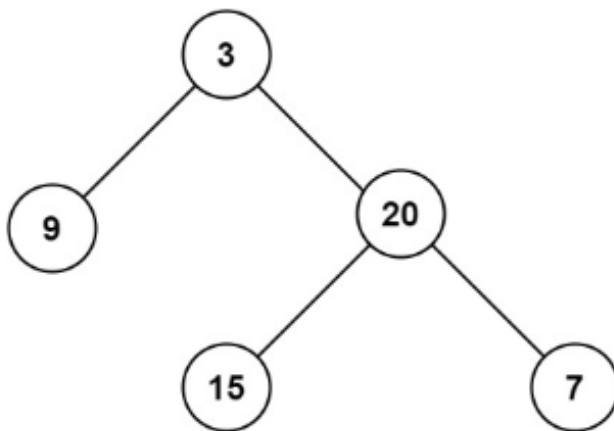
104. Maximum Depth of Binary Tree

Easy 11134 179 Add to List Share

Given the `root` of a binary tree, return its *maximum depth*.

A binary tree's **maximum depth** is the number of nodes along the longest path from the root node down to the farthest leaf node.

Example 1:



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

Output: 3

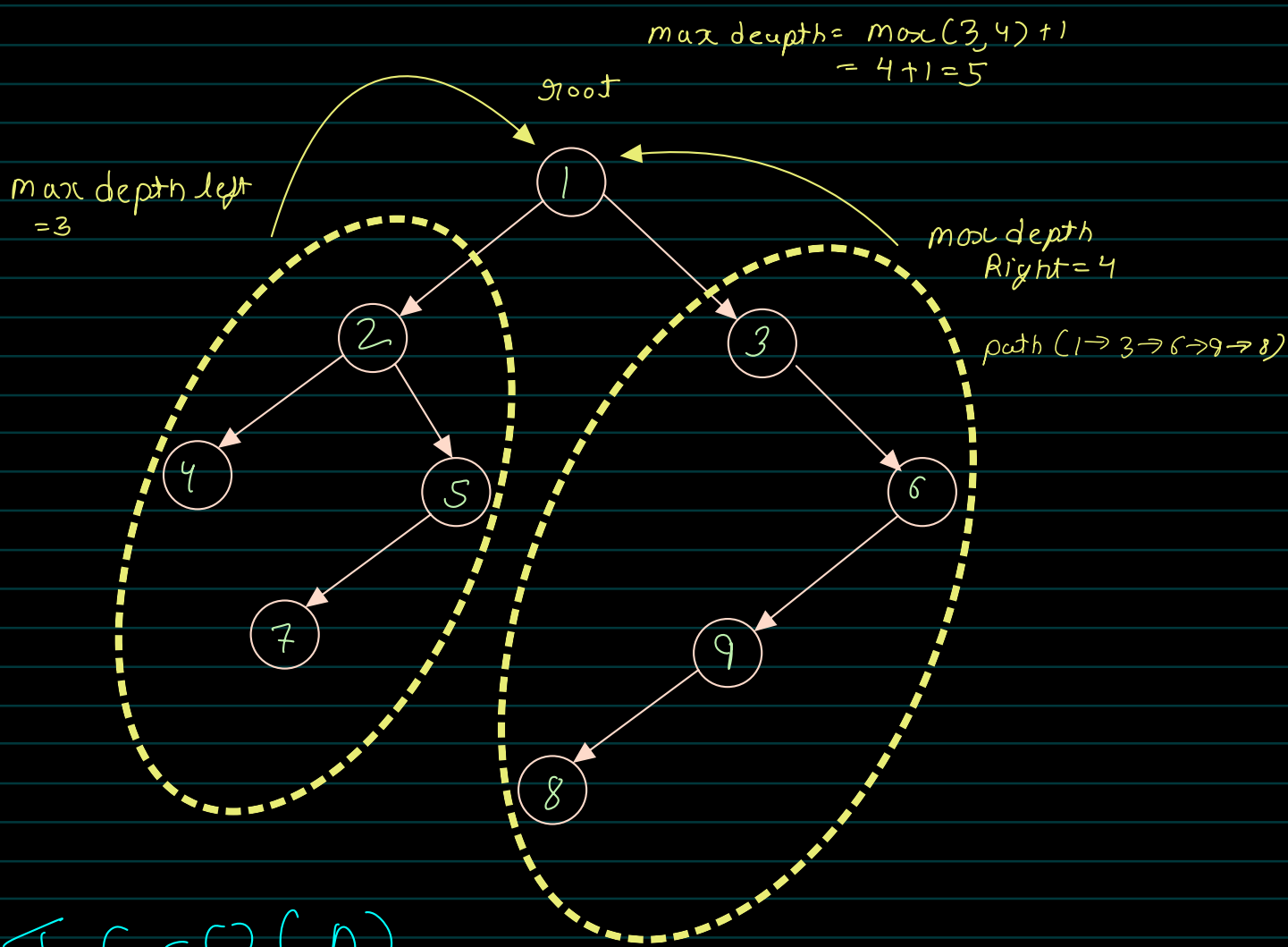
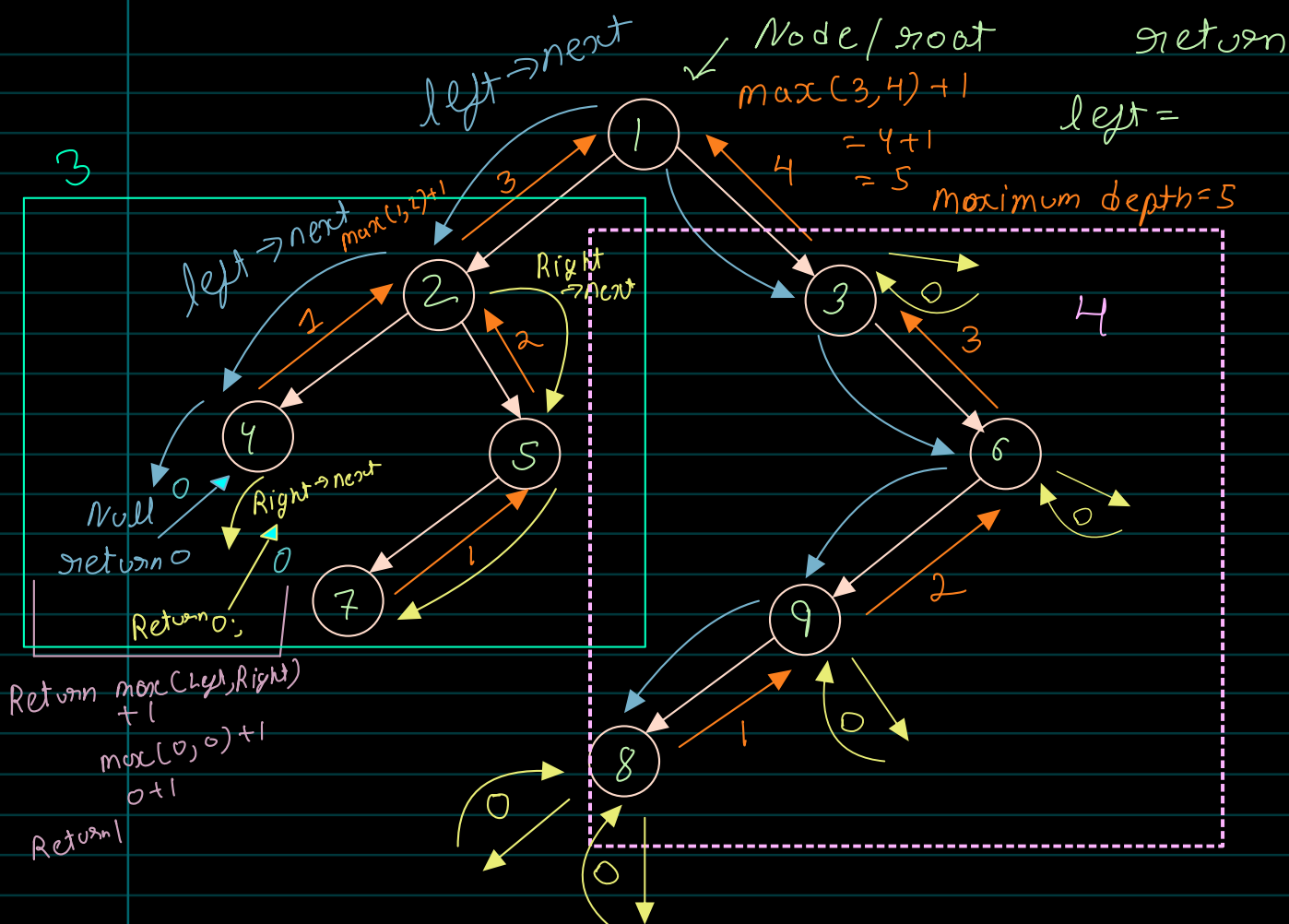
Approach:

Find the height of left sub-tree

Find the height of Right sub-tree

return $\max(\text{left_height}, \text{Right_height}) + 1$

Because we want root node included as well



T.C - $O(n)$

S.C - $O(1)$

Code

```
class Solution {  
public:  
    int height(TreeNode* root){  
        if(root==NULL) return 0;  
        int left=height(root->left);  
        int right=height(root->right);  
  
        return max(left,right)+1;  
    }  
    int maxDepth(TreeNode* root) {  
        int h=height(root);  
        return h;  
    }  
};
```