

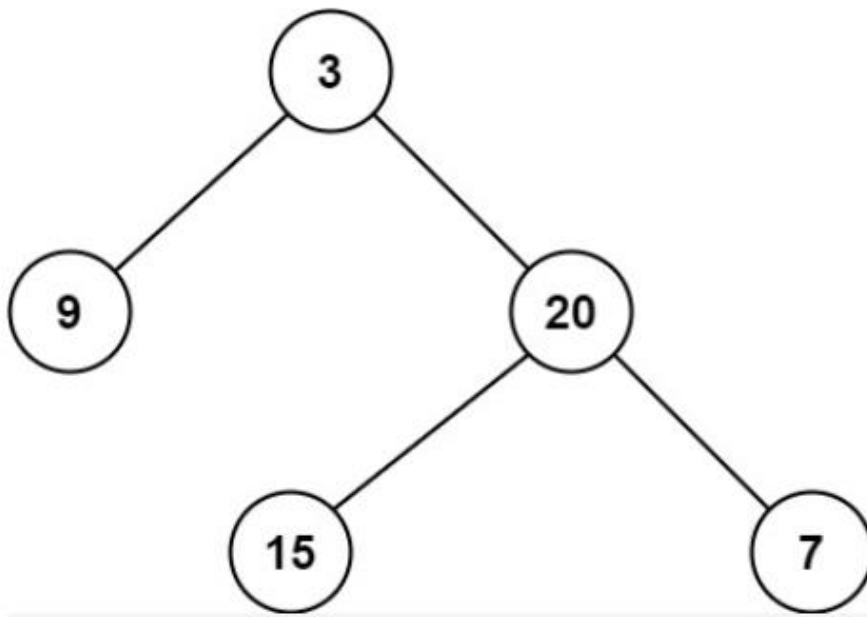
二叉树的最小深度

给定一个二叉树，找出其最小深度。

最小深度是从根节点到最近叶子节点的最短路径上的节点数量。

说明：叶子节点是指没有子节点的节点。

示例 1：



输入：root = [3,9,20,null,null,15,7]

输出：2

示例 2：

输入：root = [2,null,3,null,4,null,5,null,6]

输出：5

```
/**
 * Definition for a binary tree node.
 * struct TreeNode {
 *     int val;
 *     TreeNode *left;
 *     TreeNode *right;
 *     TreeNode() : val(0), left(nullptr), right(nullptr) {}
 *     TreeNode(int x) : val(x), left(nullptr), right(nullptr) {}
 *     TreeNode(int x, TreeNode *left, TreeNode *right) : val(x), left(left), right(right) {}
 * };
 */
```

```
class Solution {
public:

    int minDepth(TreeNode* root) {
        if(root==nullptr)
            return 0;
        if(root->left==nullptr&&root->right==nullptr)
        {
            return 1;
        }
        if(root->left==nullptr)
        {
            return 1+minDepth(root->right);
        }
        if(root->right==nullptr)
        {
            return 1+minDepth(root->left);
        }
        return 1+min(minDepth(root->left),minDepth(root->right));
    }
};
```