LeetCode

Given a binary tree, find its minimum depth.

The minimum depth is the number of nodes along the shortest path from the root node down to the nearest leaf node.

**Note:** A leaf is a node with no children.

/\*\*

\* 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==NULL)

return 0;

if(root->left==NULL && root->right==NULL)

return 1;

if(root->left==NULL)

return minDepth(root->right)+1;

if(root->right==NULL)

return minDepth(root->left)+1;

return min(minDepth(root->left),minDepth(root->right))+1;

}

};