Maximum Depth of Binary Tree

Description:

Given a binary tree, find its maximum depth.

The maximum depth is the number of nodes along the longest path from the root node down to the farthest leaf node.

Instances:

|  |  |  |
| --- | --- | --- |
|  | | maximum depth = 0 |
|  | | maximum depth = 1 |
|  |  | maximum depth = 2 |
|  | | maximum depth = 2 |

Code – Recursive:

/\*\*

\* Definition for a binary tree node.

\* struct TreeNode {

\* int val;

\* TreeNode \*left;

\* TreeNode \*right;

\* TreeNode(int x) : val(x), left(NULL), right(NULL) {}

\* };

\*/

class Solution {

public:

int maxDepth(TreeNode\* root) {

if (root == NULL) {

return 0;

}

return max(maxDepth(root->left),maxDepth(root->right)) + 1;

}

};