题目难度：MEDIUM

给出一个完全二叉树，求出该树的节点个数。

说明：

完全二叉树的定义如下：在完全二叉树中，除了最底层节点可能没填满外，其余每层节点数都达到最大值，并且最下面一层的节点都集中在该层最左边的若干位置。若最底层为第 h 层，则该层包含 1~ 2h 个节点。

示例:

输入:

1

/ \

2 3

/ \ /

4 5 6

输出: 6

思路：树+DFS

先判断空树返回0

直接DFS板子，每次执行累加一次结果

最后返回结果即可

时间击败56%

空间击败22%

上代码：

/\*\*

\* 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 res = 0;

void DFS(TreeNode\* node) {

res++;

if(node->left == nullptr && node->right == nullptr) return ;

if(node->left != nullptr) DFS(node->left);

if(node->right != nullptr) DFS(node->right);

}

int countNodes(TreeNode\* root) {

if(root == nullptr) return res;

DFS(root);

return res;

}

};