111 Minimum Depth of Binary Tree

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.

就是根到叶的最短路径

这个题比较简单了，每次我们只需要取左右子树中的短高度即可。

需要注意的是，如果只有左节点，没有右节点。其实它的高度只能是左节点了。如，

```
    2
   /
  1
```

这时的高度应该是2，而不是1.

```
/**
 * 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:
    /// 辅助函数，求一棵树的最小高度
    int depth(TreeNode* root){
        if(root == NULL)  return 0;

        int l = depth(root->left);
        int r = depth(root->right);
        int sub_depth;
        if( l == 0 ) sub_depth = r;
        if( r == 0 ) sub_depth = l;
        if( l !=0 && r!= 0 ) sub_depth = min(l,r);

        return sub_depth + 1;
    }

    int minDepth(TreeNode* root) {
        if( root == NULL )  return 0;
        return depth(root);        // 1 是包括根的整棵树的高度
    }
};
```