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.

Solution 1

```
public class Solution {
    public int minDepth(TreeNode root) {
        if(root == null) return 0;
        int left = minDepth(root.left);
        int right = minDepth(root.right);
        return (left == 0 || right == 0) ? left + right + 1: Math.min(left,right)
+ 1;

    }
}
```

written by caiqi8877 original link here

Solution 2

```
class Solution {
public:
    int minDepth(TreeNode *root) {
        if(!root) return 0;
        if(!root->left) return 1 + minDepth(root->right);
        if(!root->right) return 1 + minDepth(root->left);
        return 1+min(minDepth(root->left),minDepth(root->right));
    }
};
```

written by wulinjiansheng original link here

Solution 3

Why expected result for input of {1,2} is 2? Shouldn't it be 1?

written by ruilingrandy original link here