110. Balanced Binary Tree

Add to List | Question | Editorial Solution | My Submissions (/problems/balanced-binary-tree/submissions/)

Total Accepted: **152024** Total Submissions: **418972** Difficulty: **Easy** Contributors: **Admin**

Given a binary tree, determine if it is height-balanced.

For this problem, a height-balanced binary tree is defined as a binary tree in which the depth of the two subtrees of *every* node never differ by more than 1.

Subscribe (/subscribe/) to see which companies asked this question

Show Tags

Show Similar Problems

Have you met this question in a real interview? Yes No

Discuss (https://discuss.leetcode.com/category/118) | Top Solutions | Pick One (/problems/random-one-question/)

C++

```
/**
 * 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:
    bool isBalanced(TreeNode* root) {
        return helper(root,0)>=0;
    }
private:
    int helper(TreeNode* root, int depth){
        if(root==NULL) return depth;
        int ld = helper(root->left,depth+1);
        if(ld==-1) return -1;
        int rd = helper(root->right,depth+1);
        if(rd==-1) return -1;
        if(abs(ld-rd)>1) return -1;
        return max(ld,rd);
    }
};
```

Custom Testcase

Contribute Testcase

Run Code | Submit Solution

Notes

Send Feedback (mailto:admin@leetcode.com?subject=Feedback)