110. Balanced Binary Tree ★

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

Total Accepted: **135335** Total Submissions: **381012** Difficulty: **Easy**

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.

Notes

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://leetcode.com/discuss/questions/oj/balanced-binary-tree)

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:
    int checkHeight(TreeNode* root) {
        if (!root) return 0;
        int left = checkHeight(root->left);
        int right = checkHeight(root->right);
        // TODO 加入違法條件
        if (left == -1 || right == -1 || abs(left - right) > 1) return -1;
        //
        return max(left, right)+1;
    }

    bool isBalanced(TreeNode* root) {
        return checkHeight(root) >= 0;
    }
};
```

Custom Testcase

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

Run Code | Submit Solution

Notes