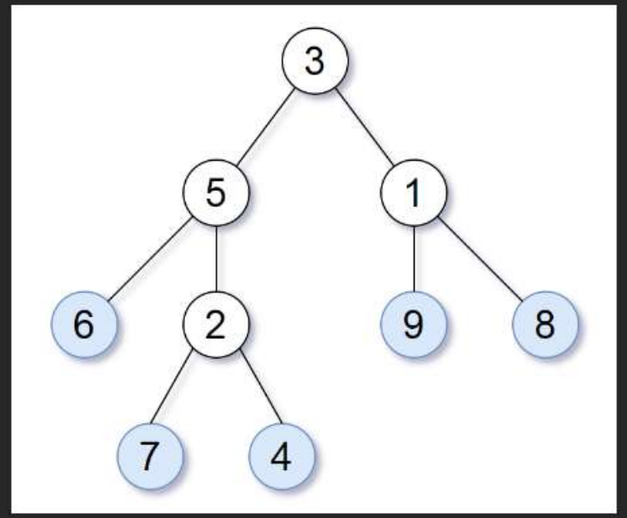
Description | Editorial | Solutions | Submissions

872. Leaf-Similar Trees

Solved

Easy | Topics | Companies

Consider all the leaves of a binary tree, from left to right order, the values of those leaves form a **leaf value sequence**.



For example, in the given tree above, the leaf value sequence is `(6, 7, 4, 9, 8)`.

Two binary trees are considered *leaf-similar* if their leaf value sequence is the same.

Return `true` if and only if the two given trees with head nodes `root1` and `root2` are leaf-similar.

Example 1:

4K | 87

Code

C | Auto

```c
/**
 * Definition for a binary tree node.
 * struct TreeNode {
 *     int val;
 *     struct TreeNode *left;
 *     struct TreeNode *right;
 * };
 */
void getLeafSequence(struct TreeNode* root, int *index, int* leaves) {
    if (root == NULL) {
        return;
    }

    if (root->left == NULL && root->right == NULL) {
        leaves[(*index)++] = root->val;
    }

    getLeafSequence(root->left, index, leaves);
    getLeafSequence(root->right, index, leaves);
}

bool leafSimilar(struct TreeNode* root1, struct TreeNode* root2) {
    int leaves1[100] = {0};
    int leaves2[100] = {0};

    int index1 = 0;
    int index2 = 0;

    getLeafSequence(root1, &index1, leaves1);
    getLeafSequence(root2, &index2, leaves2);

    if (index1 != index2) {
```

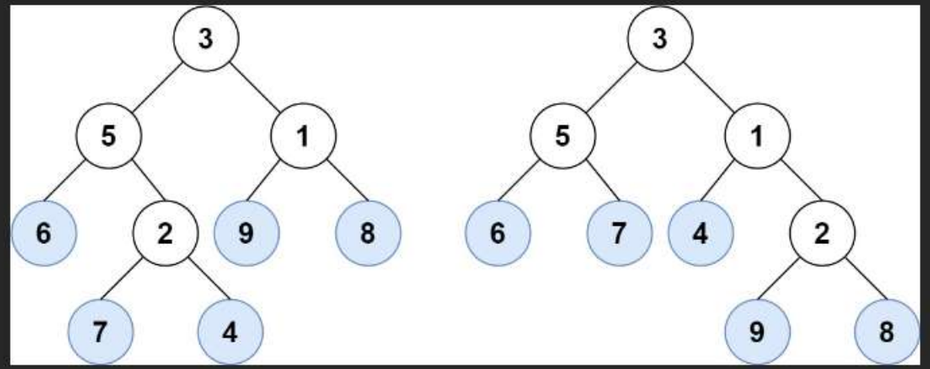
Ln 1, Col 1 | Saved to local

Run | Submit

Testcase | Test Result

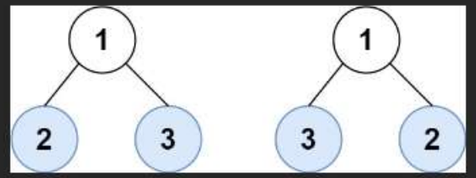
Description | Editorial | Solutions | Submissions

Example 1:



Input: root1 = [3,5,1,6,2,9,8,null,null,7,4], root2 = [3,5,1,6,7,4,2,null,null,null,null,null,null,9,8]
Output: true

Example 2:



Input: root1 = [1,2,3], root2 = [1,3,2]
Output: false

Constraints:

- The number of nodes in each tree will be in the range `[1, 200]`.

Code

C | Auto

```c

    getLeafSequence(root->left, index, leaves);
    getLeafSequence(root->right, index, leaves);
}

bool leafSimilar(struct TreeNode* root1, struct TreeNode* root2) {
    int leaves1[100] = {0};
    int leaves2[100] = {0};

    int index1 = 0;
    int index2 = 0;

    getLeafSequence(root1, &index1, leaves1);
    getLeafSequence(root2, &index2, leaves2);

    if (index1 != index2) {
        return false;
    }

    for (int i = 0; i < index1; i++) {
        if (leaves1[i]!= leaves2[i]) {
            return false;
        }
    }

    return true;
}

```

Ln 1, Col 1 | Saved to local

Run | Submit

Testcase | Test Result

Testcase | Test Result

Accepted Runtime: 0 ms

• Case 1 • Case 2

Input

root1 =

[3,5,1,6,2,9,8,null,null,7,4]

root2 =

[3,5,1,6,7,4,2,null,null,null,null,null,null,9,8]

Output

true

Expected

true

Contribute a testcase

Code

Testcase | Test Result

Accepted Runtime: 0 ms

• Case 1 • Case 2

Input

root1 =

[1,2,3]

root2 =

[1,3,2]

Output

false

Expected

false

Contribute a testcase