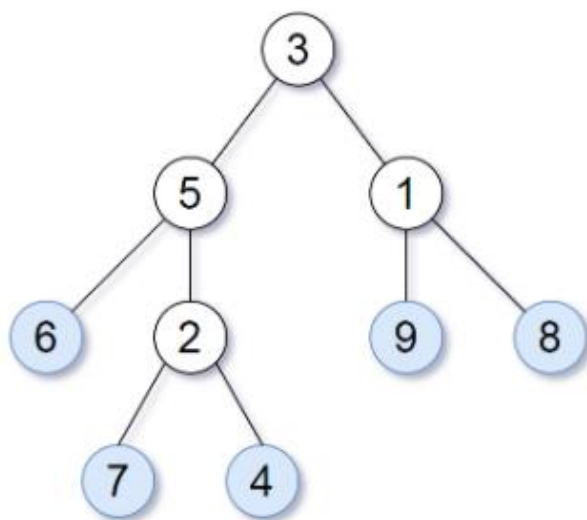


叶子相似的树

请考虑一棵二叉树上所有的叶子，这些叶子的值按从左到右的顺序排列形成一个 *叶值序列*。

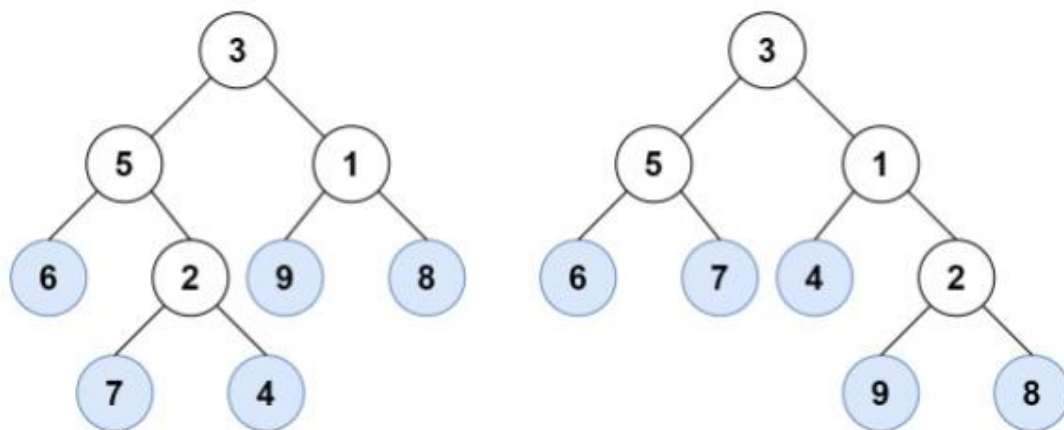


举个例子，如上图所示，给定一棵叶值序列为 (6, 7, 4, 9, 8) 的树。

如果有两棵二叉树的叶值序列是相同，那么我们就认为它们是 *叶相似* 的。

如果给定的两个头结点分别为 root1 和 root2 的树是叶相似的，则返回 true；否则返回 false。

示例 1:



输入: 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,9,8]

输出: true

```

/**
 * Definition for a binary tree node.
 * struct TreeNode {
 *     int val;
 *     TreeNode *left;
 *     TreeNode *right;
 *     TreeNode() : val(0), left(nullptr), right(nullptr) {}
 *     TreeNode(int x) : val(x), left(nullptr), right(nullptr) {}
 *     TreeNode(int x, TreeNode *left, TreeNode *right) : val(x), left(left
), right(right) {}
 * };
 */
class Solution {
public:
    void travel(TreeNode* root, vector<int>& v)
    {
        if(root==nullptr)
            return;
        if(root->left==nullptr&&root->right==nullptr)
            v.push_back(root->val);
        travel(root->left,v);
        travel(root->right,v);
    }
    bool leafSimilar(TreeNode* root1, TreeNode* root2) {
        vector<int> v1;
        vector<int> v2;
        travel(root1,v1);
        travel(root2,v2);
        if(v1.size()!=v2.size())
        {
            return false;
        }
        else
        {
            return v1==v2;
        }
    }
};

```