

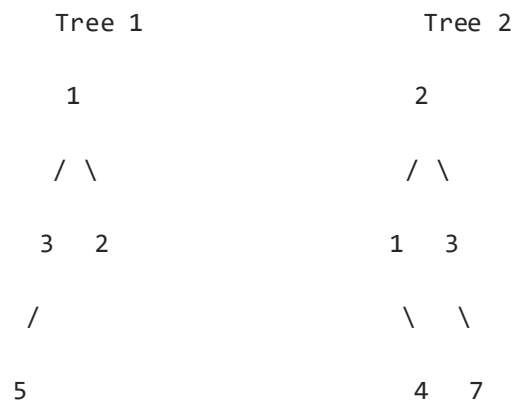
合并二叉树

给定两个二叉树，想象当你将它们中的一个覆盖到另一个上时，两个二叉树的一些节点便会重叠。

你需要将他们合并为一个新的二叉树。合并的规则是如果两个节点重叠，那么将他们的值相加作为节点合并后的新值，否则**不为 NULL** 的节点将直接作为新二叉树的节点。

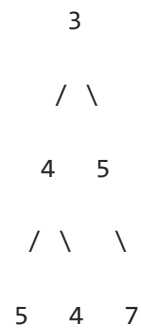
示例 1:

输入:



输出:

合并后的树:



注意: 合并必须从两个树的根节点开始。

```
/**
 * 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:
    TreeNode* mergeTrees(TreeNode* root1, TreeNode* root2) {
        if(root1==nullptr)
            return root2;
        if(root2==nullptr)
            return root1;
        TreeNode* root=new TreeNode(root1->val+root2->val);
        root->left=mergeTrees(root1->left,root2->left);
        root->right=mergeTrees(root1->right,root2->right);
        return root;
    }
};
```