두 이진 트리를 병합하라. 중복되는 노드는 값을 합산한다.

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

```
# Definition for a binary tree node.
# class TreeNode:
#     def __init__(self, val=0, left=None, right=None):
#         self.val = val
#         self.left = left
#         self.right = right
```

1.dfs

```
class Solution:
    def mergeTrees(self, root1: TreeNode, root2: TreeNode) -> TreeNode:
        def dfs(node1, node2):
            if node1 == None and node2 == None:
                return None
            if node1 == None:
                node1 = TreeNode()
            if node2 == None:
                node2 = TreeNode()
            node1.left = dfs(node1.left, node2.left)
            node1.right = dfs(node1.right, node2.right)
            val = node1.val + node2.val
            node1.val = val
            return node1

        root1 = dfs(root1, root2)
        return root1
```

```
class Solution:
    def mergeTrees(self, root1: TreeNode, root2: TreeNode) -> TreeNode:
        if root1 and root2:
            node = TreeNode(root1.val + root2.val)
            node.left = self.mergeTrees(root1.left, root2.left)
            node.right = self.mergeTrees(root1.right, root2.right)
            return node
        else:
            return root1 or root2
```