**Justin\_Tree\_0872.**  **Leaf-Similar Trees**

**Concept:**

找到每個 son 節點，並加入 list

left 要比 right 先找

最後比較兩個 list 是否相等

**Code:**

# 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

class Solution:

def leafSimilar(self, root1: TreeNode, root2: TreeNode) -> bool:

result1 = []

result2 = []

def find(node, result\_list):

if node.left and node.right:

find(node.left, result\_list)

find(node.right, result\_list)

elif node.left:

find(node.left, result\_list)

elif node.right:

find(node.right, result\_list)

else:

result\_list.append(node.val)

find(root1, result1)

find(root2, result2)

return result1 == result2