Target sum for binary tree

Link: <https://www.acwing.com/problem/content/45/>

# Definition for a binary tree node.

# class TreeNode(object):

# def \_\_init\_\_(self, x):

# self.val = x

# self.left = None

# self.right = None

class Solution(object):

def findPath(self, root, sum):

"""

:type root: TreeNode

:type sum: int

:rtype: List[List[int]]

"""

def rec(root,ret,res,tol):

if root:

#print tol

if not root.left and not root.right:

tol-=root.val

if not tol:

res+=" "+str(root.val)

ret.append([int(ele) for ele in res.split(" ")[1:]])

return

if tol<=0:

return

res+=" "+str(root.val)

tol-=root.val

rec(root.left,ret,res,tol)

rec(root.right,ret,res,tol)

ret,res = [],""

rec(root,ret,res,sum)

return ret