def maxHeight(root):

    if root is None:

        return 0

    return max(maxHeight(root.left),maxHeight(root.right)) + 1

class ListNode:

    def \_\_init\_\_(self, val, left = None, right = None) -> None:

        self.val = val

        self.left = None

        self.right = None

def heightBal(root)->bool:

    if root is None:

        return True

    if root.left is not None and root.right is not None:

        if abs(maxHeight(root.left)-maxHeight(root.right)) > 1:

            return False

        n = heightBal(root.left)

        if not n:

            return n

        else:

            return heightBal(root.right)

    elif root.left is None:

        if maxHeight(root.right) <= 1:

            return True

        return False

    elif root.right is None:

        if maxHeight(root.left) <= 1:

            return True

        return False

    else:

        return True

a1 = ListNode(3)

a1.left = ListNode(9)

a1.right = ListNode(20)

a1.right.left = ListNode(15)

a1.right.right = ListNode(7)

# a1.right.right.left = ListNode(29)

print(heightBal(a1))