# Definition for singly-linked list.

# class ListNode:

# def \_\_init\_\_(self, val=0, next=None):

# self.val = val

# self.next = next

class Solution:

def addTwoNumbers(self, l1: ListNode, l2: ListNode) -> ListNode:

tempSum = 0

# will return temp.next

temp = ListNode(0)

# this will be used to traverse the linkedList and keep adding the sums from L1 and L2

curr = temp

#carry over to be added

carry = 0

while l1 or l2 or carry !=0:

val1 = l1.val if l1 else 0

val2 = l2.val if l2 else 0

carry, out = divmod(val1+val2+carry,10)

curr.next = ListNode(out)

curr = curr.next

l1 = (l1.next if l1 else None)

l2 = (l2.next if l2 else None)

return temp.next