给出两个 **非空** 的链表用来表示两个非负的整数。其中，它们各自的位数是按照 **逆序** 的方式存储的，并且它们的每个节点只能存储 **一位** 数字。

如果，我们将这两个数相加起来，则会返回一个新的链表来表示它们的和。

您可以假设除了数字 0 之外，这两个数都不会以 0 开头。

**示例：**

**输入：**(2 -> 4 -> 3) + (5 -> 6 -> 4)

**输出：**7 -> 0 -> 8

**原因：**342 + 465 = 807

ListNode\* addTwoNumbers(ListNode\* l1, ListNode\* l2) {

if (!l1) return l2;

if (!l2) return l1;

ListNode \*headDummy = new ListNode(-1);

ListNode \*tail = headDummy, \*p = l1, \*q = l2;

int flag = 0;

while (p || q)

{

int value1 = p ? p->val : 0;

int value2 = q ? q->val : 0;

int value = (value1 + value2 + flag) % 10;

flag = (value1 + value2 + flag) / 10;

ListNode \*node = new ListNode(value);

tail->next = node;

tail = node;

if (p) p = p->next;

if (q) q = q->next;

}

if (flag == 1)

{

ListNode \*node = new ListNode(1);

tail->next = node;

tail = node;

}

return headDummy->next;

}