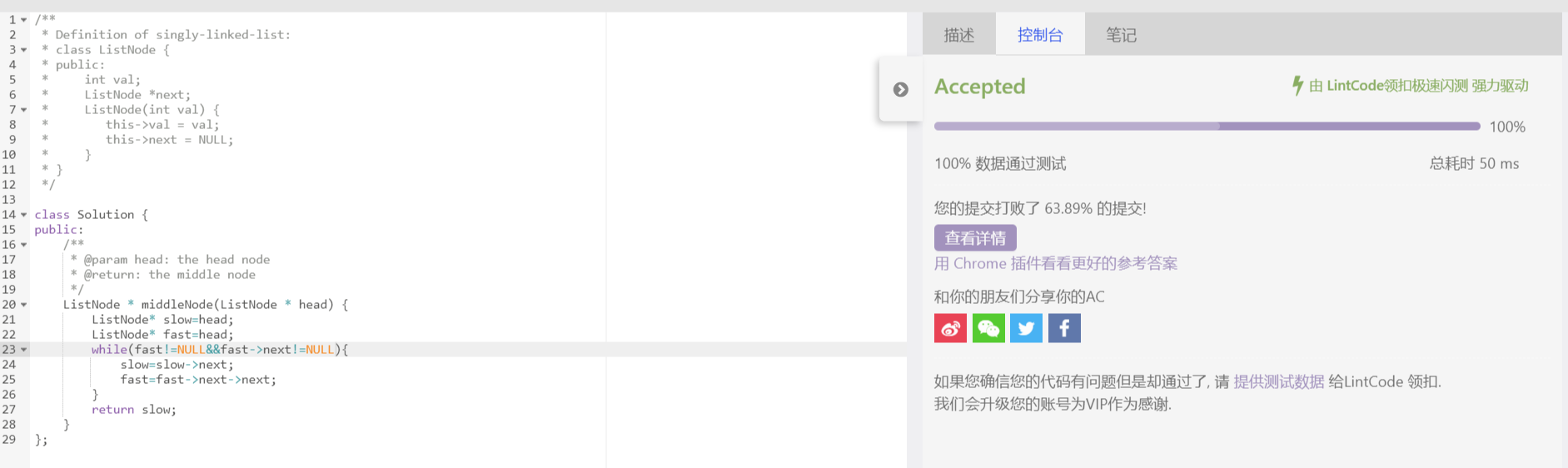
**2019级软工暑期算法训练**

**11J191 周晗**

**一、链表**

1609 给定一个带有头结点 head 的非空单链表，返回链表的中间结点。如果有两个中间结点，则返回第二个中间结点。



实现思路：用快慢指针的思想，快指针是慢指针速度的两倍，快指针到列表尾时，慢指针恰好位于中间。

822 给出一个链表，并将链表的值以倒序存储到数组中。



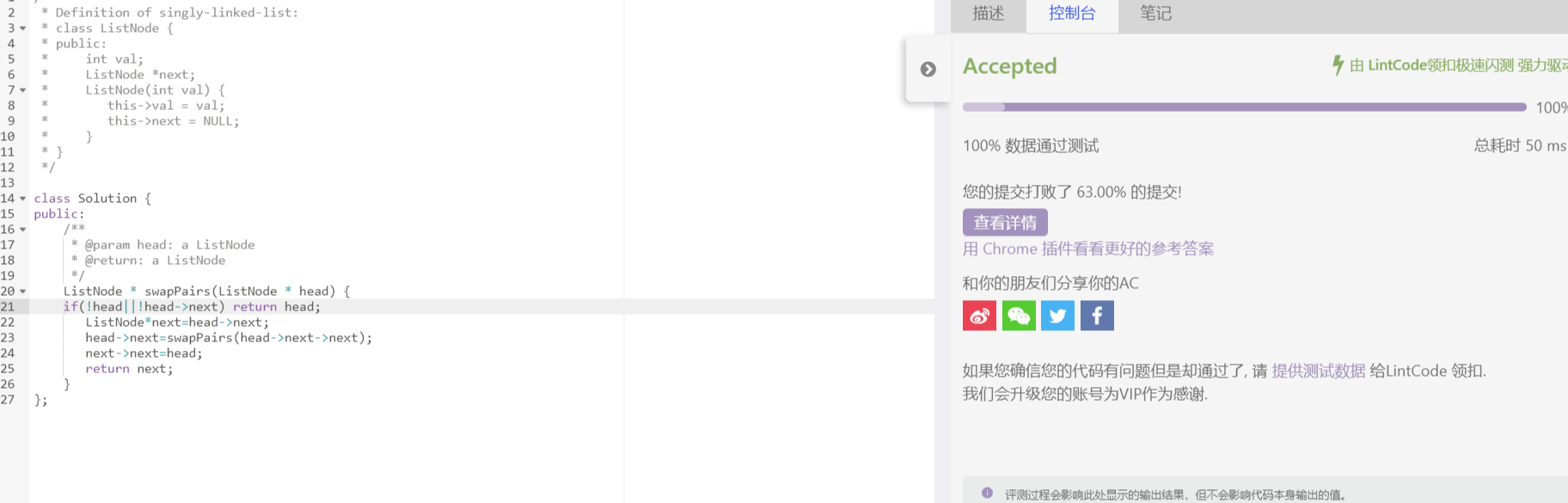
实现思路：不确定数组长度，用vector，配合push-back输入数据，最后反转就可以。

756 给出两个链表形式表示的数字,写一个函数得到这两个链表相乘乘积。



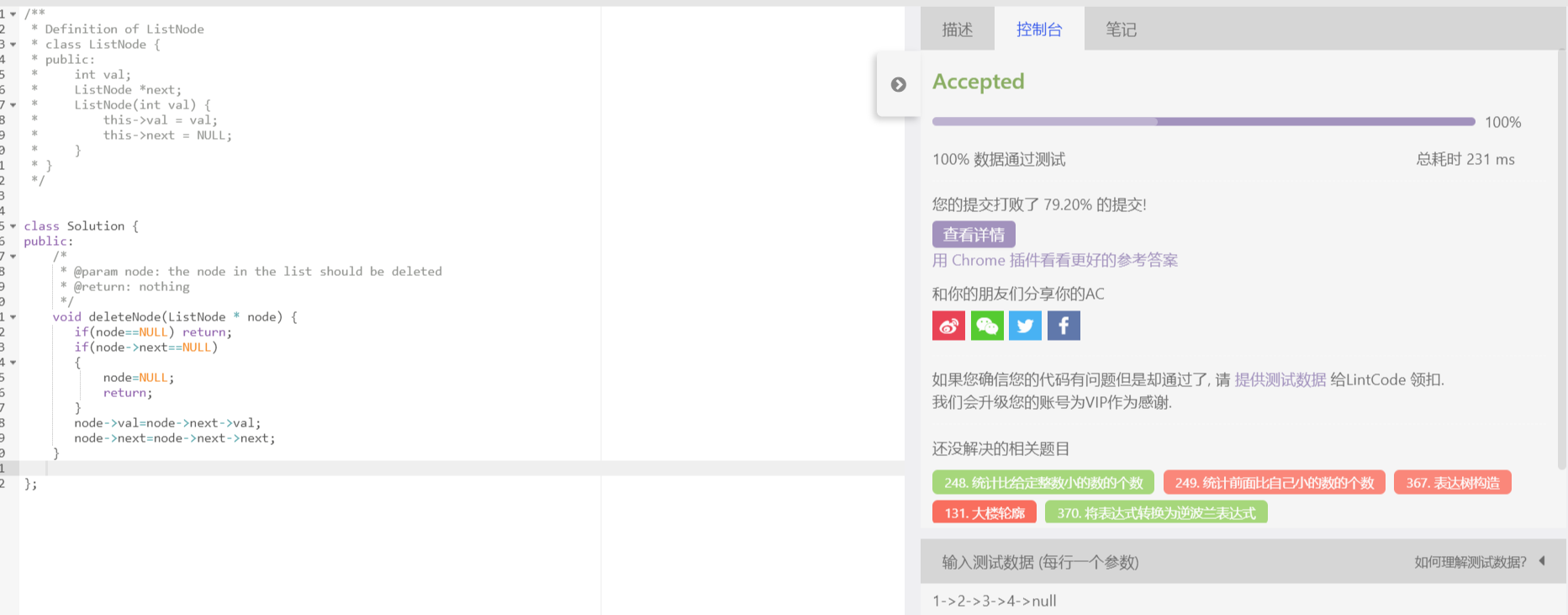
实现思路：按照链表遍历得到两数相乘返回结果。

451 给一个链表，两两交换其中的节点，然后返回交换后的链表。

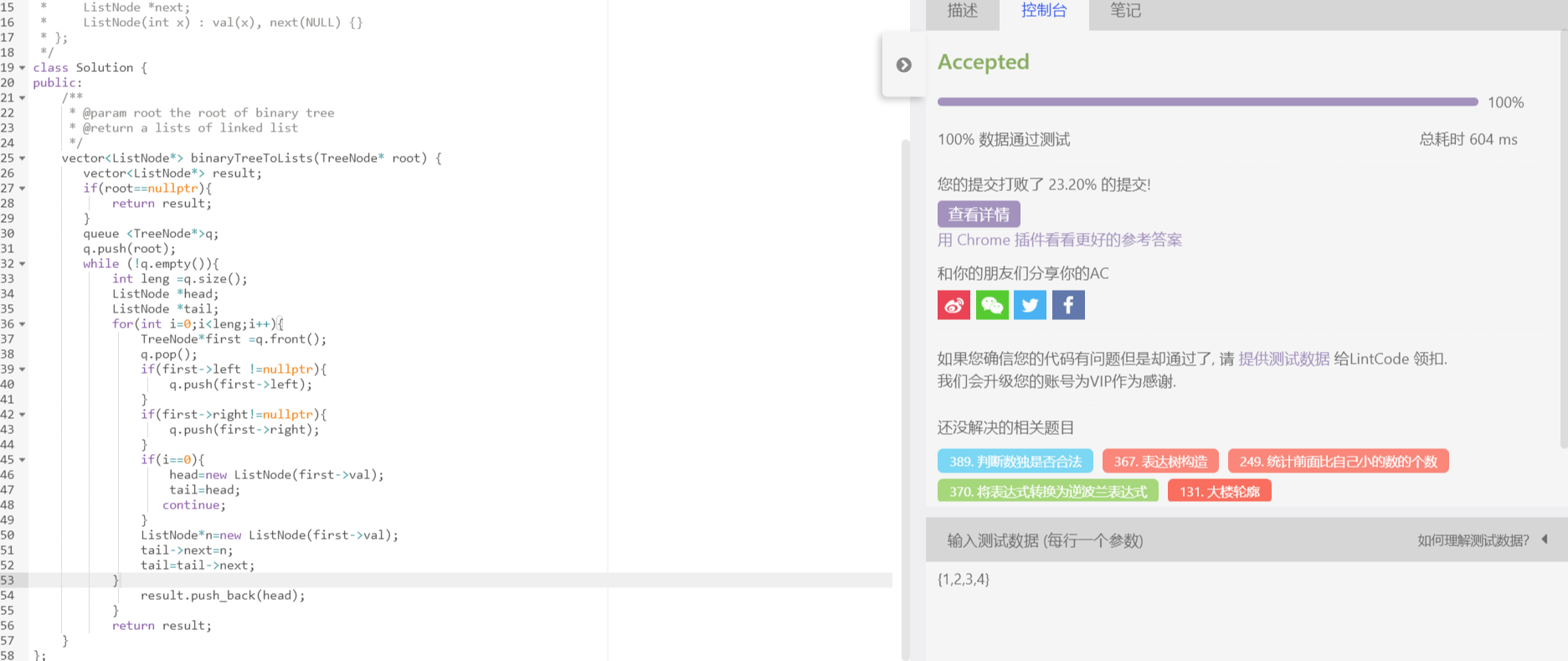


实现思路：递归，先设定递归退出条件（到最后剩下一个空节点或是单个节点），然后保存第二个节点，将第三个节点进行递归，返回结果被第一个节点指向，然后之前保存的第二个节点指向第一个节点，返回第二个节点即可。

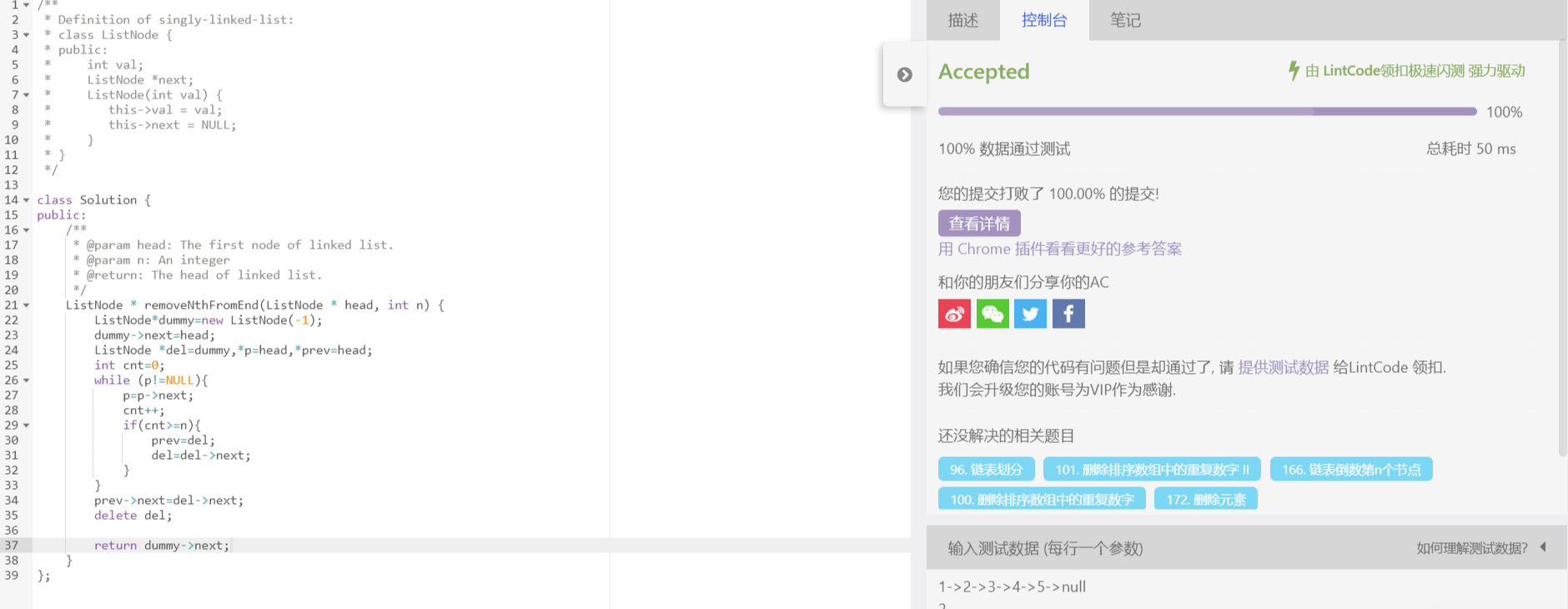
372 给定一个单链表中的一个等待被删除的节点(非表头或表尾)。请在在 O(1) 时间复杂度删除该链表节点。



实现思路：把之前删除的node的值改成下一个node的值，它的next改成下一个next的值即可。

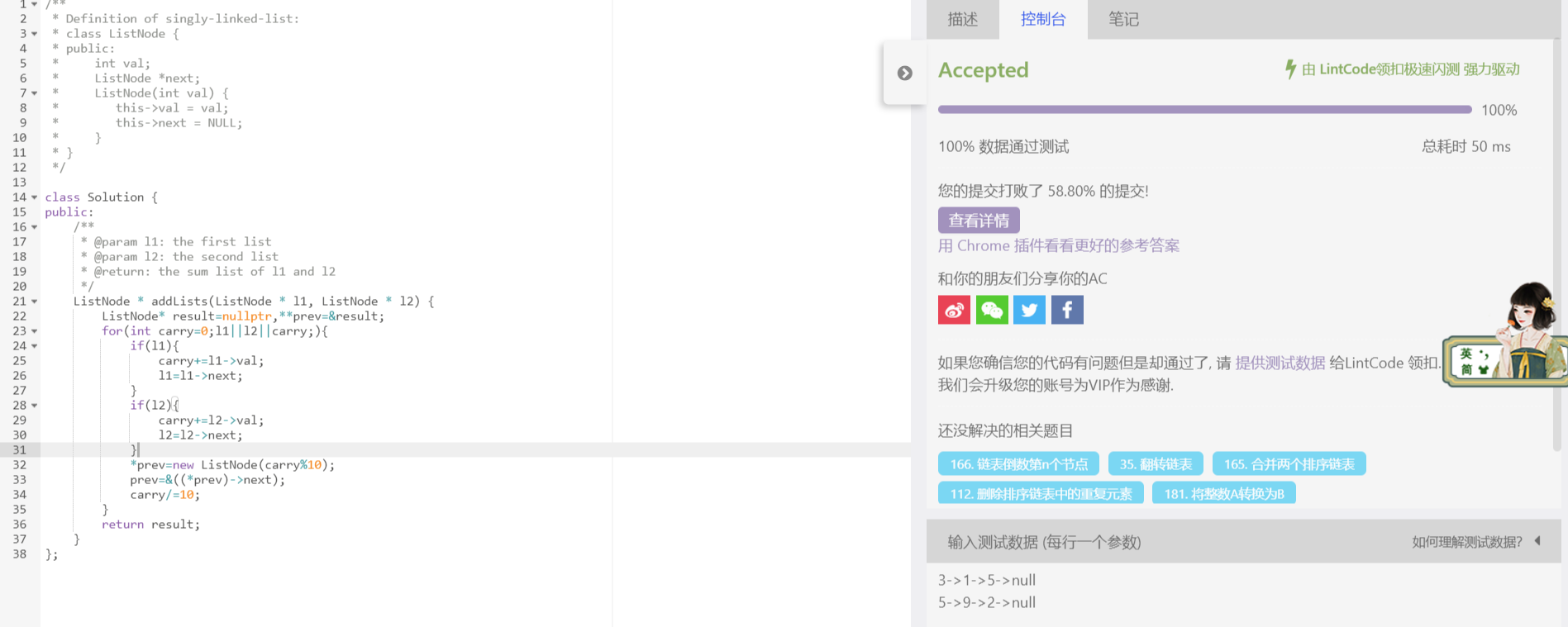
242 给一棵二叉树，设计一个算法为每一层的节点建立一个链表。也就是说，如果一棵二叉树有 D 层，那么你需要创建 D 条链表。

174 给定一个链表，删除链表中倒数第n个节点，返回链表的头节点。



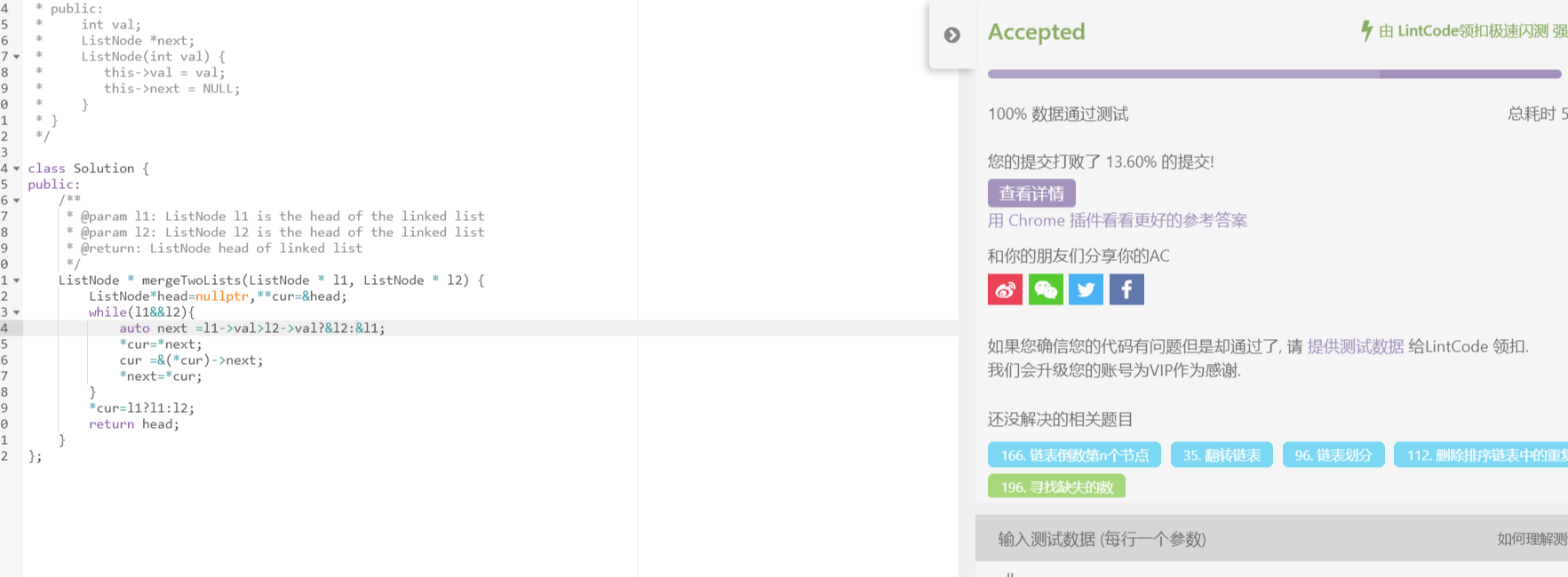
实现思路：头指针先跑n下，待删指针再开始动。

167 你有两个用链表代表的整数，其中每个节点包含一个数字。数字存储按照在原来整数中相反的顺序，使得第一个数字位于链表的开头。写出一个函数将两个整数相加，用链表形式返回和。



实现思路：快慢指针， 快指针先n步拉开距离， 然后快慢指针同时走， 快指针到头， 慢速指针即为目标。

165 将两个排序链表合并为一个新的排序链表



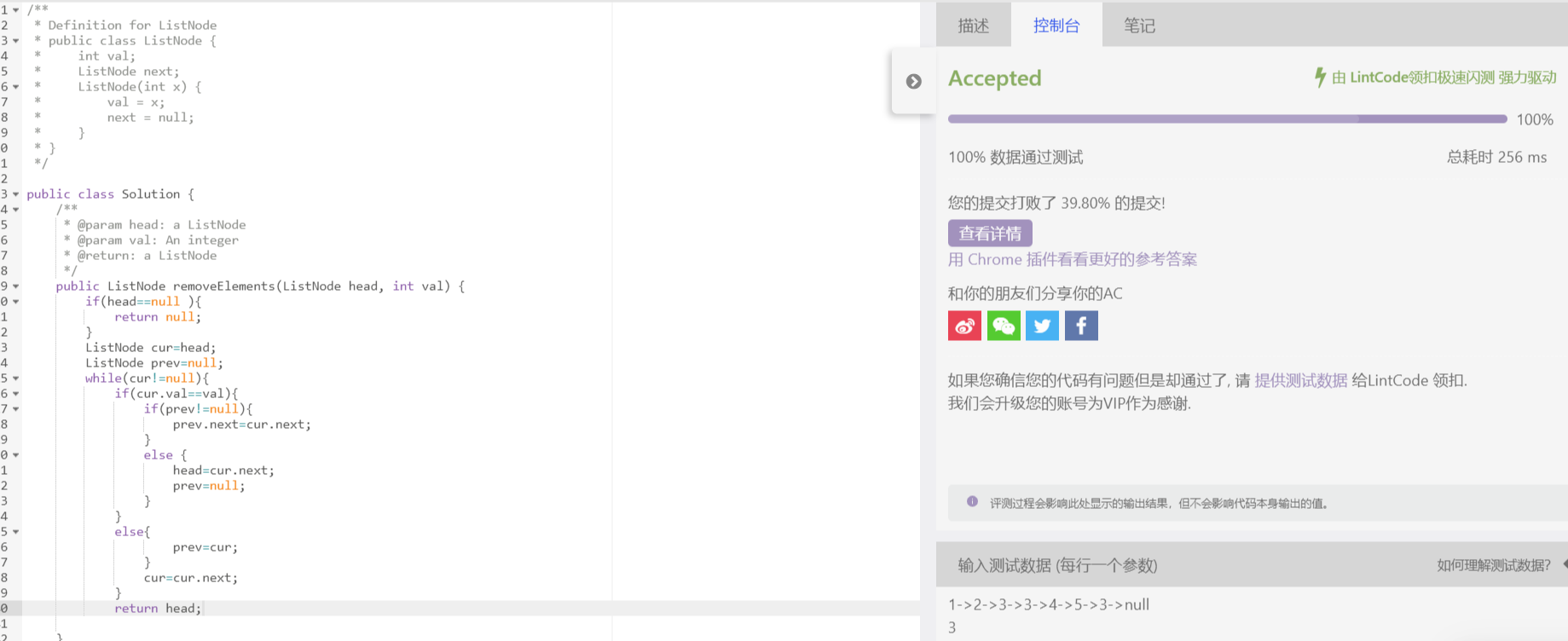
实现思路： 先建一个dummy node,用cur指针指向dummy，循环链表，条件是：while l1 and l2，比较l1 和 l2 , 并和cur.next 链接，同时更新l1 和 l2 的下个节点，最后返回dummy。

112 给定一个排序链表，删除所有重复的元素每个元素只留下一个。



实现思路：删除重复元素，不跳过它们。用一个cur指针指向head，然后遍历即可。

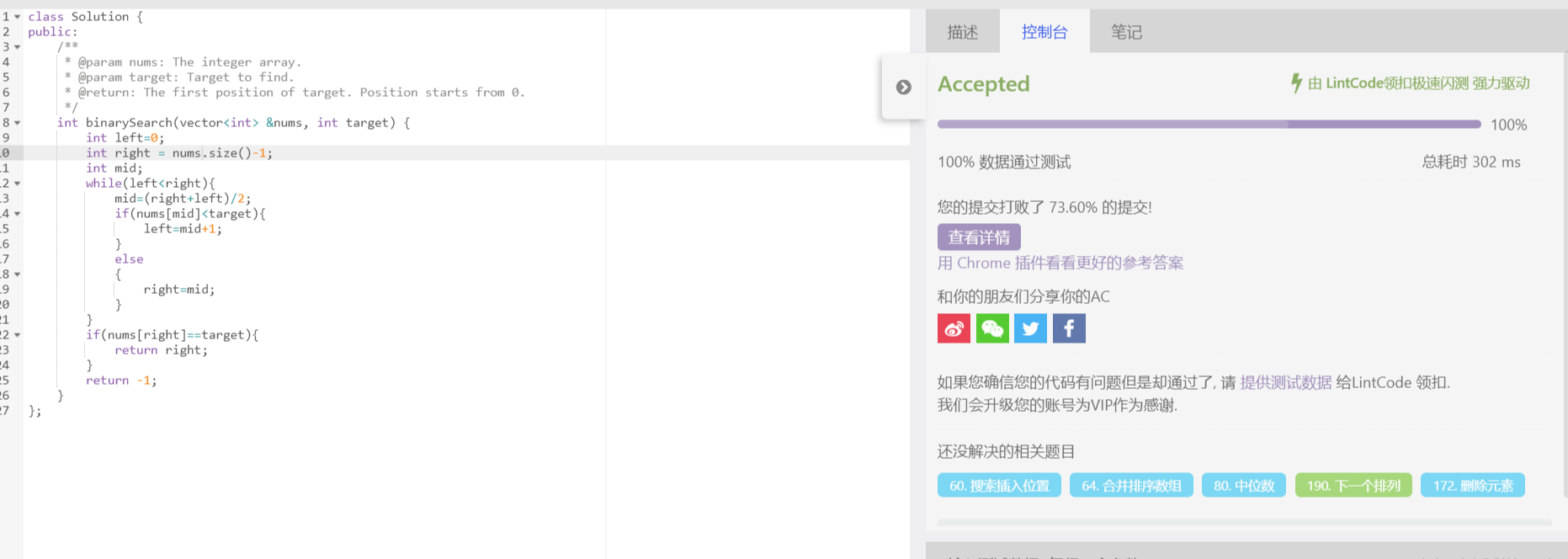
452 删除链表中等于给定值 val 的所有节点。



实现思路：用prev和cur；两个指针指向之前节点和当前节点，如果当前是头结点，则prev为null，即考虑要删的是不是头节点。

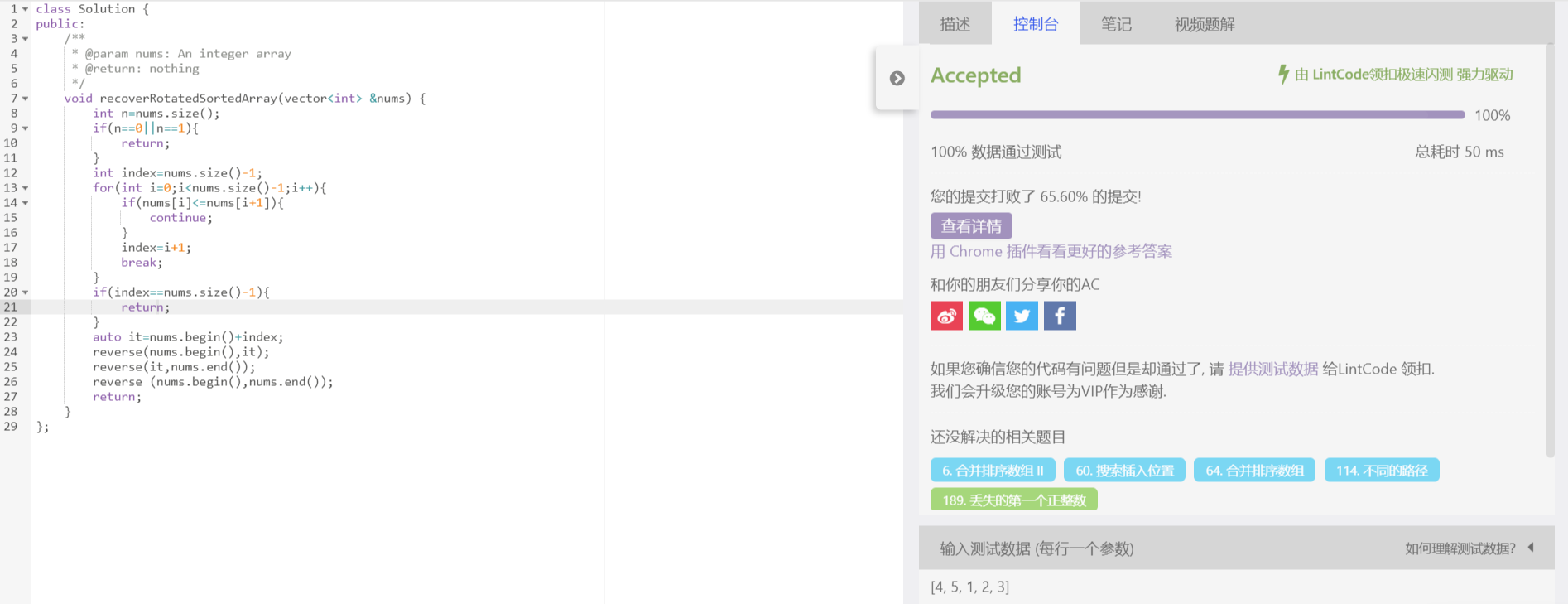
**二、数组**

14. 给定一个排序的整数数组（升序）和一个要查找的整数target，用O(logn)的时间查找到target第一次出现的下标（从0开始），如果target不存在于数组中，返回-1。



实现思路：将区间赋值为整个数组区间，去中间位置mid，若mid<target，则将区间缩小到原区间的右区间，mid>=target时，同理。若left>=right时，nums[right]=target则返回right，否则返回-1。

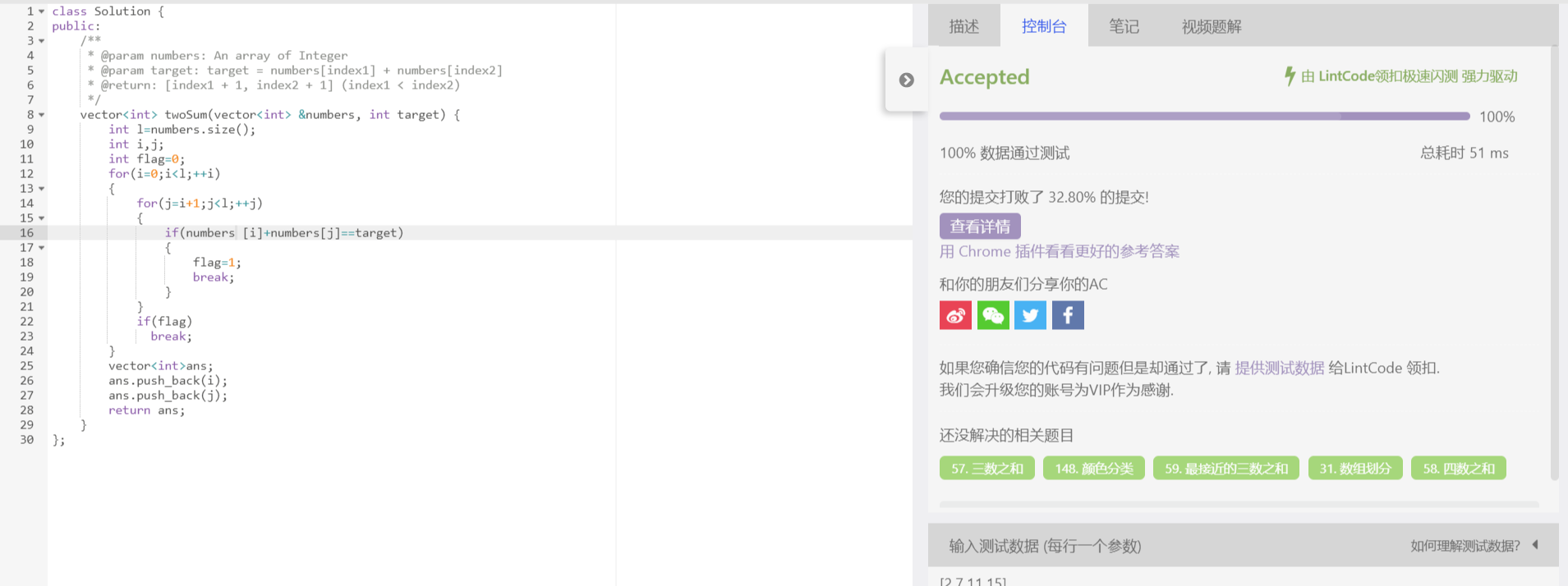
39. 给定一个旋转排序数组，在原地恢复其排序。（升序）。



实现思路：借鉴三步翻转法。

56. 给一个整数数组，找到两个数使得他们的和等于一个给定的数 target。

你需要实现的函数twoSum需要返回这两个数的下标, 并且第一个下标小于第二个下标。注意这里下标的范围是 0 到 n-1。



实现思路：暴力枚举每一个i和j，如果加起来和为target即为答案，falg作为找到答案后跳出的第一个标记。

60. 给定一个排序数组和一个目标值，如果在数组中找到目标值则返回索引。如果没有，返回到它将会被按顺序插入的位置。你可以假设在数组中无重复元素。



实现思路：二分法，缩小target在数组A的范围，优先判断数组长度是否为0.

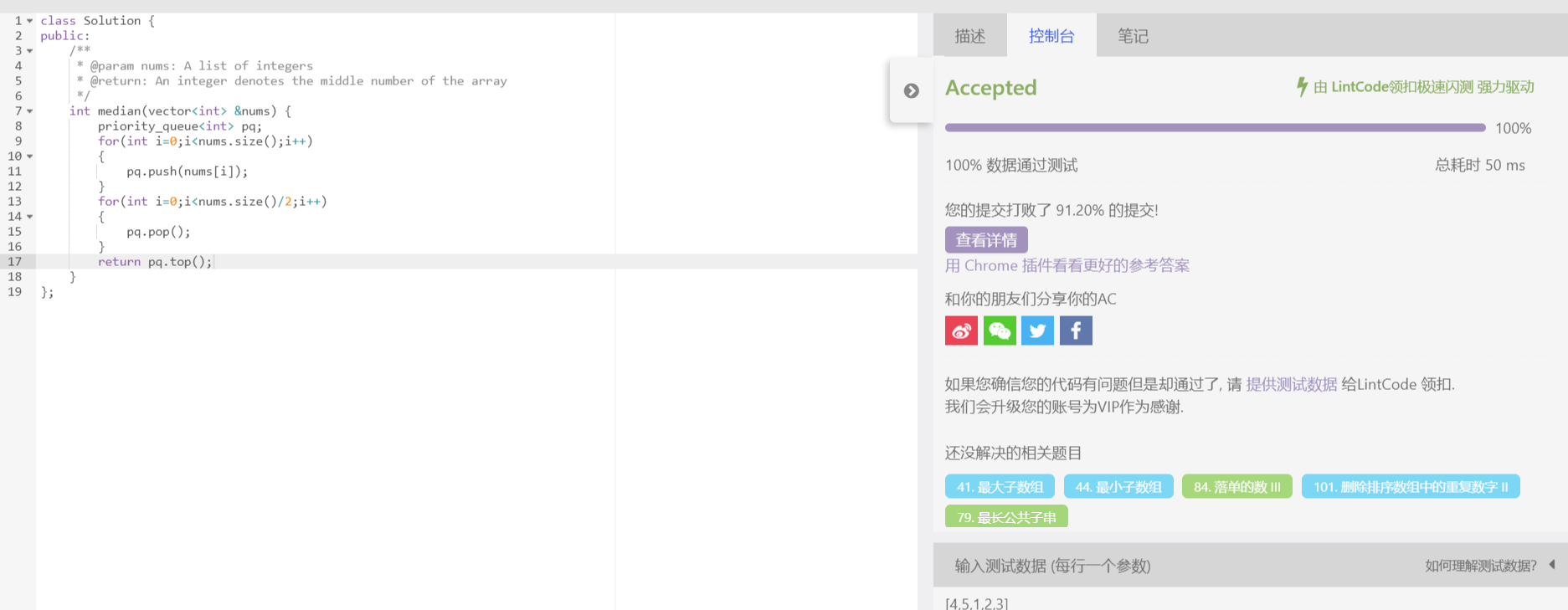
64. 合并两个排序的整数数组A和B变成一个新的数组。



实现思路：从A的最后一个元素下标（m+n-1）开始，向前走，若B先用完，直接return，若A先用完，把B剩余的元素放在A前。

80. 给定一个未排序的整数数组，找到其中位数。

中位数是排序后数组的中间值，如果数组的个数是偶数个，则返回排序后数组的第N/2个数。



实现思路：本题借鉴了lintcode上的方法，使用priority queue将所有数排序，再依次取出N/2个数。

100. 给定一个排序数组，在原数组中“删除”重复出现的数字，使得每个元素只出现一次，并且返回“新”数组的长度。

不要使用额外的数组空间，必须在不使用额外空间的条件下原地完成。



实现思路:快慢指针，如果相同就把快指针后移，慢指针不动，否则同时后移。

101.

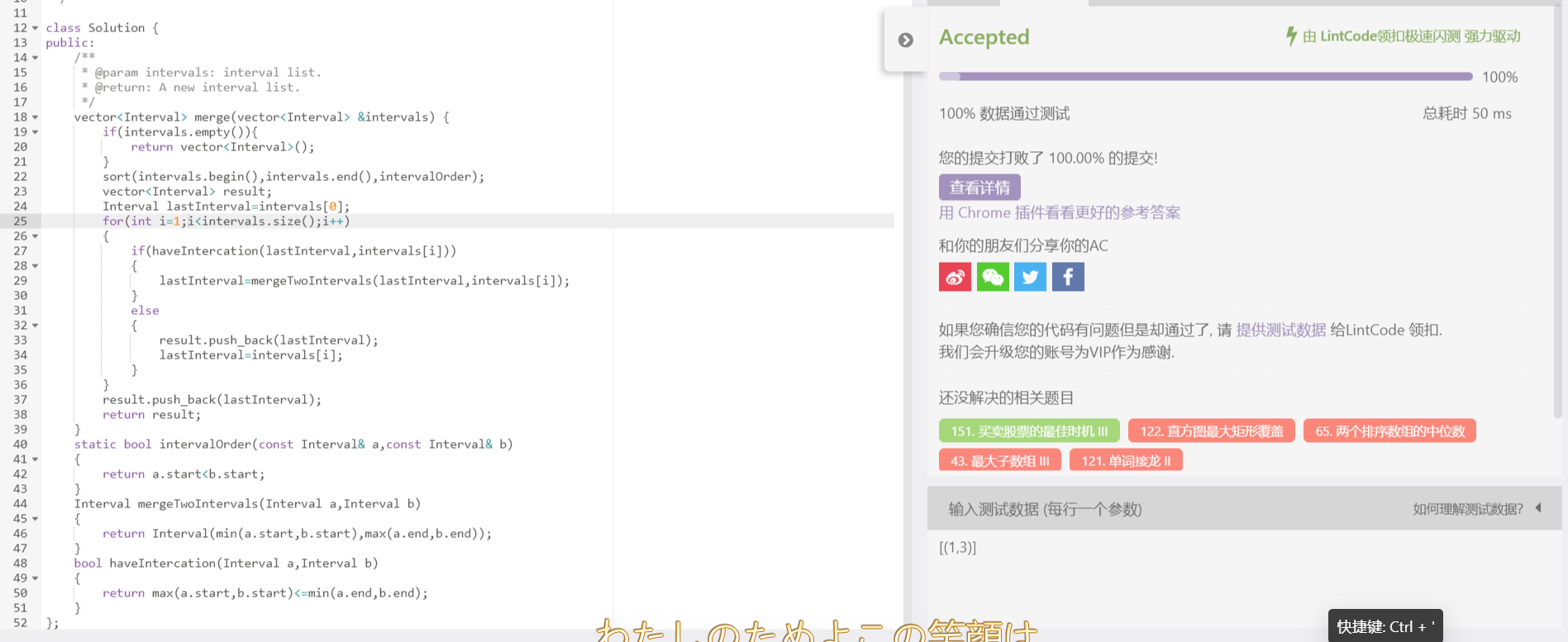
给你一个排序数组，删除其中的重复元素，使得每个数字最多出现两次，返回新的数组的长度。

如果一个数字出现超过2次，则这个数字最后保留两个。



实现思路：与上题类似，使用快慢指针，主要区别在于如果有数字重复，则将慢指针向右移动一次。

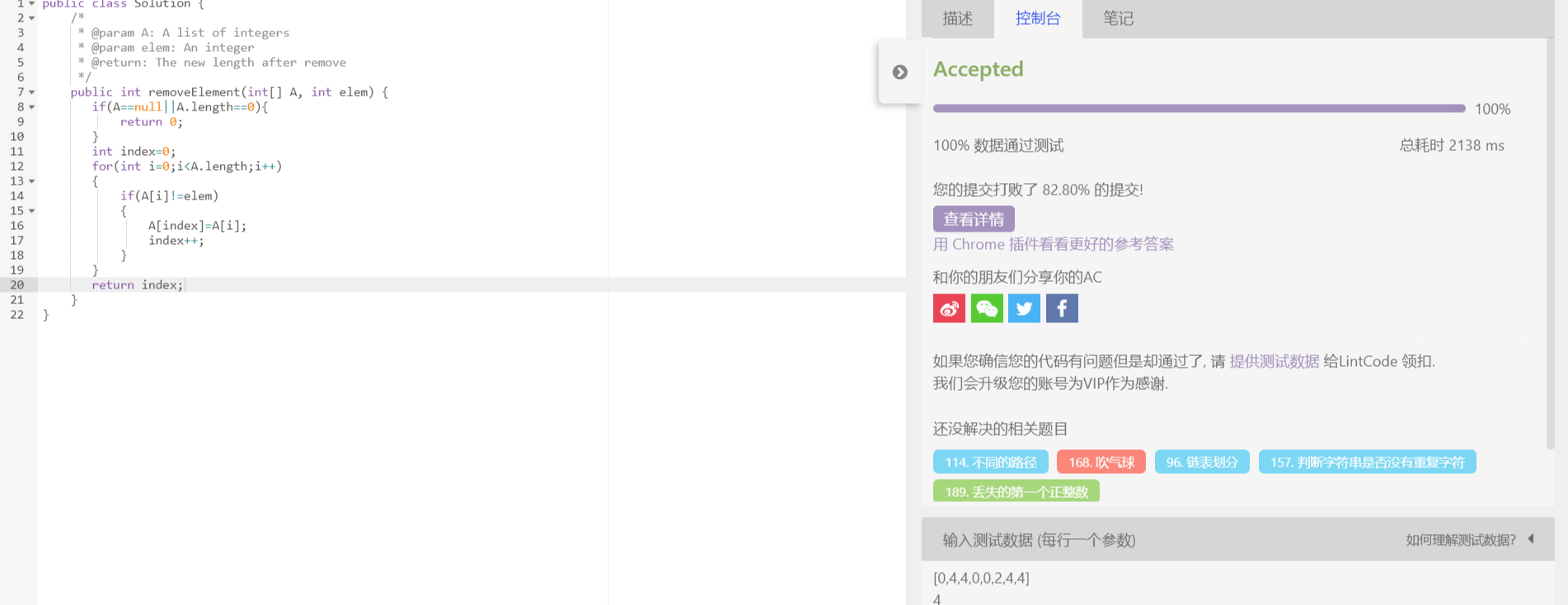
156. 给出若干闭合区间，合并所有重叠的部分。



实现思路：对区间数组按区间左端点start排序，将最后的区间赋值lastInterval为intervals【0】，遍历输入，如果lastinterval和当前区间相交，合并两个区间，如果不相见，将lastinterval加入结果，并将lastinterval赋值为当前的区间。

172. 给定一个数组和一个值，在原地删除与值相同的数字，返回新数组的长度。

元素的顺序可以改变，并且对新的数组不会有影响。

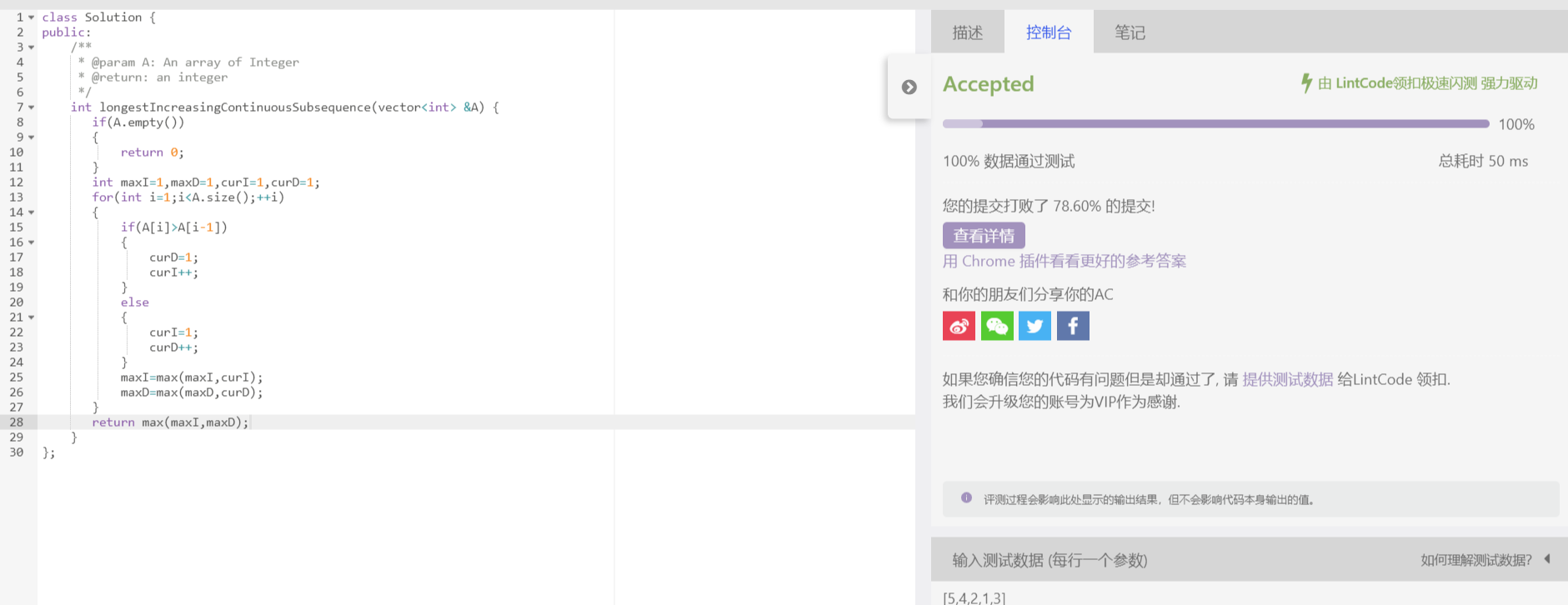


实现思路：并没有考虑到两个指针，遍历一遍就过了。

280. 在一个二维平面上有许多城市，所有的城市都有着自己的名字c[i]c[i]，以及位置坐标(x[i],y[i])(x[i],y[i])（都为整数）。有qq组询问，每组询问给出一个城市名字，你需要回答离该城市距离最近的，xx相同或者yy相同的城市名称。

实现思路：本题没能完成。

397. 给定一个整数数组（下标从 0 到 n-1， n 表示整个数组的规模），请找出该数组中的最长上升连续子序列。（最长上升连续子序列可以定义为从右到左或从左到右的序列。）



实现思路：同时记录上升和下降序列，发生序列逆转是，重设长度为1。

407. 给定一个非负数，表示一个数字数组，在该数的基础上+1，返回一个新的数组。

该数字按照数位高低进行排列，最高位的数在列表的最前面。



实现思路：从数组尾开始遍历，如果数字是9则将数字赋值0，并继续向前遍历；直到数字不是9或者遍历到第一个数字。如果不是9，则该数字加一后退出遍历，若加一后的第一个元素是0（证明存在进位），给结果数组添加一个“1”，最后把其他元素依次加入到数组中。

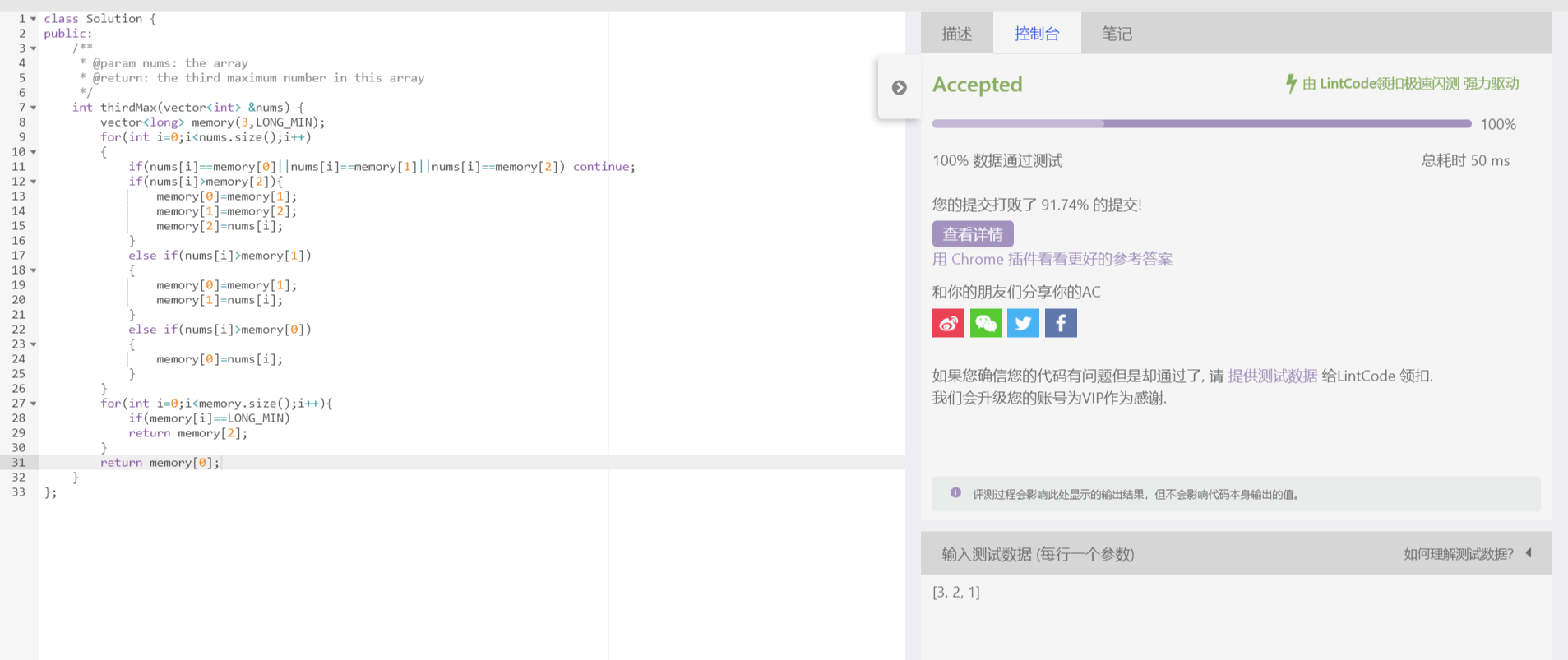
1157. 给定一个整数数组，你需要找到一个连续子数组，如果你只按升序对这个子数组进行排序，那么整个数组也将按升序排序。

你需要找到最短的这样的子数组并输出它的长度。



实现思路：对整个原序列排序，用排序之后的序列与原序列的首尾进行一一对比，出去首尾一一对应相等的元素之后剩下的序列即为最短的无序子序列。

1250. 给定一个非空数组，返回此数组中第三大的数。如果不存在，则返回数组中最大的数。要求算法时间复杂度必须是O(n)。



实现思路：用三个变量来存放第一大，第二大，第三大的元素的变量，然后遍历数组，最后得到第三大的元素，若没有则返回第一大的元素。