1. 题目
2. 题干

给定一个排序数组，你需要在 原地 删除重复出现的元素，使得每个元素只出现一次，返回移除后数组的新长度。

不要使用额外的数组空间，你必须在 原地 修改输入数组 并在使用 O(1) 额外空间的条件下完成。

1. 示例

给定数组 nums = [1,1,2],

函数应该返回新的长度 2, 并且原数组 nums 的前两个元素被修改为 1, 2。

你不需要考虑数组中超出新长度后面的元素。

1. 题解
2. 思路

题目要求原地删除重复出现的元素，组成新的数组，返回新数组的长度。由于，只能在原地删除重复的元素，不能添加额外的空间，因此直接循环数组并创建新数组的方式是行不通的。不能添加新的空间，联想到使用双指针的方式，一个是慢指针i，一个是快指针j，只要nums[i]==nums[j]，那么就j++跳过重复项；若nums[i]!=nums[j]，则两者不相同，那么将nums[j]的值复制到nums[i+1]上，直到j到达数组的末尾。

1. 代码实现
2. Java

public class List26 {  
 public int removeDuplicates(int[] nums){  
 if(nums.length==0)  
 return 0;  
 int i = 0;  
 for(int j = 1; j < nums.length; ++j){  
 if(nums[j] != nums[i]){  
 i++;  
 nums[i] = nums[j];  
 }  
 }  
 return i+1;  
 }  
}