# 26、删除排序数组中的重复项（Remove Duplicates from Sorted Array）

## 题目：

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。

**示例 1:**

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。

**示例 2:**

给定 nums = [*0*,*0*,*1*,*1*,*1*,*2*,*2*,*3*,*3*,*4*],函数应该返回新的长度 *5*, 并且原数组 nums 的前五个元素被修改为 *0*, *1*, *2*, *3*, 4。你不需要考虑数组中超出新长度后面的元素。

**说明:**

为什么返回数值是整数，但输出的答案是数组呢?

请注意，输入数组是以「引用」方式传递的，这意味着在函数里修改输入数组对于调用者是可见的。

你可以想象内部操作如下:

// nums 是以“引用”方式传递的。也就是说，不对实参做任何拷贝 *int* len = *removeDuplicates*(nums);// 在函数里修改输入数组对于调用者是可见的。// 根据你的函数返回的长度, 它会打印出数组中该长度范围内的所有元素。for (*int* i = *0*; i < len; i++) { *print*(nums[i]);}

## 解答：

|  |
| --- |
| public static int RemoveDuplicates(int[] nums)  {  //举例0,0,1,1,1,2,2,3,3,4  //定义两个变量i和p，p始终指向第一个有效的点，i点负责遍历整个数组，当i到达的位置的值和p对应的值  //相同时，p不变，i继续往下找。当两个值相同时先把p往后移一位，再把i对应的值赋给p  int p = 0;  for (int i = 1; i < nums.Length; i++)  {  if (nums[i] != nums[p])  {  p++;  nums[p] = nums[i];  }  }  //别忘了为0时，会报错  if (nums.Length == 0)  {  return 0;  }  return p + 1;  } |