Remove Duplicates from Sorted Array

首先读清题意，题目很简单，删除数组（sorted）中重复的元素，然后返回唯一元素的大小。A数组要变化。

分两种情况：

1》若数族大小为0/1。 数组A 不变，返回0/1。

2》 若数组大小大于1。用count记录新数组元素A的末尾的下一个index，若a[i] != a[i-1], 则把 a[i]的值赋给count位置上（a[count] = a[i] )。

一开始只想着从原数组上去覆盖了（链表思想），末尾的Index找的出了问题，附上了一段错误的代码：

class Solution {

public:

int removeDuplicates(vector<int>& nums) {

int l;

int count=0;

l=nums.size();

if(l==0||l==1){

return l;

}

for(int i=0;i<l;){

if(nums[i]==nums[i+1]){

nums[i+1-count]=nums[i+2-count];

count++;

i=i+2;

}

else{

i++;

}

}

return l-count;

}

};

后来读懂别人的思路，

代码如下：（后面的数与前面的数比较）

class Solution {

public:

int removeDuplicates(vector<int>& nums) {

int l;

int count=1;

l=nums.size();

if(l==0||l==1){

return l;

}

for(int i=1;i<l;i++){

if(nums[i]!=nums[i-1]){

nums[count]=nums[i];

count++;

}

}

return count;

}

};

又发现另一种：（前面的数与后面比较）这两种只是视野不同

class Solution {

public:

int removeDuplicates(vector<int>& nums) {

int l;

int count=1;

l=nums.size();

if(l==0||l==1){

return l;

}

for(int i=1;i<l;i++){

if(nums[count-1]!=nums[i]){

nums[count]=nums[i];

count++;

}

}

return count;

}

};