/\*

Follow up for "Remove Duplicates":

What if duplicates are allowed at most twice?

For example,

Given sorted array nums = [1,1,1,2,2,3],

Your function should return length = 5, with the first five elements of nums being 1, 1, 2, 2 and 3. It doesn't matter what you leave beyond the new length.

way-1:挨个测试，看是不是只有一个，再看是不是只有两个，把多余的去除

way-2:两个指针

\*/

class Solution {

public:

int removeDuplicates(vector<int>& nums)

{

//way-1

/\*

if(nums.size()==0)

return 0;

int pos=0;

while(true)

{

if(pos+1>=nums.size())

break;

if(nums[pos+1]!=nums[pos])

{

pos++;

continue;

}

if(pos+2>=nums.size())

break;

if(nums[pos+2]==nums[pos])

{

nums.erase(nums.begin()+pos+2);

}

else

{

pos=pos+2;

}

}

return nums.size();

\*/

//way-2

if(nums.size()==0)

return 0;

int pos=0;//pos左边存符合的,(pos,r)存不符合的，r之后存没判定的

int flag=0;

for(int r=1;r<nums.size();r++)

{

if(nums[r]!=nums[pos])

{

swap(nums[r],nums[++pos]);

flag=0;

}

else if(flag==0)

{

swap(nums[r],nums[++pos]);

flag++;

}

}

return pos+1;

}

};