Sort Colors

Given an array with *n* objects colored red, white or blue, sort them so that objects of the same color are adjacent, with the colors in the order red, white and blue.

Here, we will use the integers 0, 1, and 2 to represent the color red, white, and blue respectively.

**Note:**  
You are not suppose to use the library's sort function for this problem.

[click to show follow up.](https://oj.leetcode.com/problems/sort-colors/)

**Follow up:**  
A rather straight forward solution is a two-pass algorithm using counting sort.  
First, iterate the array counting number of 0's, 1's, and 2's, then overwrite array with total number of 0's, then 1's and followed by 2's.

Could you come up with an one-pass algorithm using only constant space?

就是一个排序题，

分别遍历每个数，记录最后一个红色的序号（初始为0），第一个蓝色的序号（初始为n-1），如果是红色，跟最后一个红色交换位置，最后一个红色索引+1，如果是蓝色跟第一个蓝色索引的数交换，蓝色索引－1

注意在与红色交换后遍历的索引可以直接＋1，因为交换的数只可能是红色或白色，不可能是蓝色，而与蓝色索引交换后则不一定。