

调整数组顺序使奇数位于偶数前面

输入一个整数数组，实现一个函数来调整该数组中数字的顺序，使得所有奇数位于数组的前半部分，所有偶数位于数组的后半部分。

示例：

输入：nums = [1,2,3,4]

输出：[1,3,2,4]

注：[3,1,2,4] 也是正确的答案之一。

```
class Solution {
public:
    vector<int> exchange(vector<int>& nums) {
        if(nums.empty() || nums.size() == 1) return nums;
        int size = nums.size();
        int start = 0, end = size - 1;
        while(start < end)
        {
            // 向后移动 start 直到找到一个偶数
            while(start < end && ((nums[start] & 1) != 0))
                start ++;
            // 向前移动 end 直到找到一个奇数
            while(start < end && ((nums[end] & 1) == 0))
                end --;
            if(start < end)//如果找到的偶数在前半部分 奇数在后半部分 则进行交换
            {
                int temp = nums[start];
                nums[start] = nums[end];
                nums[end] = temp;
            }
        }
        return nums;
    }
};
```