**难度: 简单**

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。

提示：

1 <= nums.length <= 50000

1 <= nums[i] <= 10000

来源：力扣（LeetCode）

链接：https://leetcode-cn.com/problems/diao-zheng-shu-zu-shun-xu-shi-qi-shu-wei-yu-ou-shu-qian-mian-lcof

著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。

思路:

1). 创建一个同样大小的数组res

2). 创建一个奇数下标=0和偶数下标 = 数组长度 -1;

3). 循环比较是否是偶数,如果不是偶数就奇数下标++传入res如果是偶数就偶数下标—传入res

4). 最后直接返回res

|  |
| --- |
| public int[] exchange(int[] nums) {  int[] res = new int[nums.length];  int oddIndex = 0; //奇数开始的下标  int evenIndex = nums.length - 1; //偶数开始的下标  for (int i = 0; i < nums.length; i++) {  if (nums[i] %2 == 0) {  res[evenIndex--] = nums[i];  }else {  res[oddIndex++] = nums[i];  }  }  return res;  } |