#### [189. 旋转数组](https://leetcode-cn.com/problems/rotate-array/)

给定一个数组，将数组中的元素向右移动 k 个位置，其中 k 是非负数。

示例 1:

输入: [1,2,3,4,5,6,7] 和 k = 3

输出: [5,6,7,1,2,3,4]

解释:

向右旋转 1 步: [7,1,2,3,4,5,6]

向右旋转 2 步: [6,7,1,2,3,4,5]

向右旋转 3 步: [5,6,7,1,2,3,4]

示例 2:

输入: [-1,-100,3,99] 和 k = 2

输出: [3,99,-1,-100]

解释:

向右旋转 1 步: [99,-1,-100,3]

向右旋转 2 步: [3,99,-1,-100]

说明:

尽可能想出更多的解决方案，至少有三种不同的方法可以解决这个问题。

要求使用空间复杂度为 O(1) 的 原地 算法。

解决方案1：

双重循环，

遍历数组，依次向右移动一位，

循环k次

时间复杂度O(n^2)

空间复杂度O(1)

解决方案2：

遍历数组，从倒数第k位向前遍历，依次将每个元素向后移动k位，

需要额外的空间记录原数组最后k个元素，最后将额外数组中的元素放入数组的前k位

时间复杂度O(n^2)

空间复杂度O(n)

解决方案3：

难以想到

将原数组前length - k 个元素转置，将原数组后k个元素转置

最后将整个数组转置，即为求得的结果

时间复杂度O(n)

空间复杂度O(1)