

旋转数组

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:

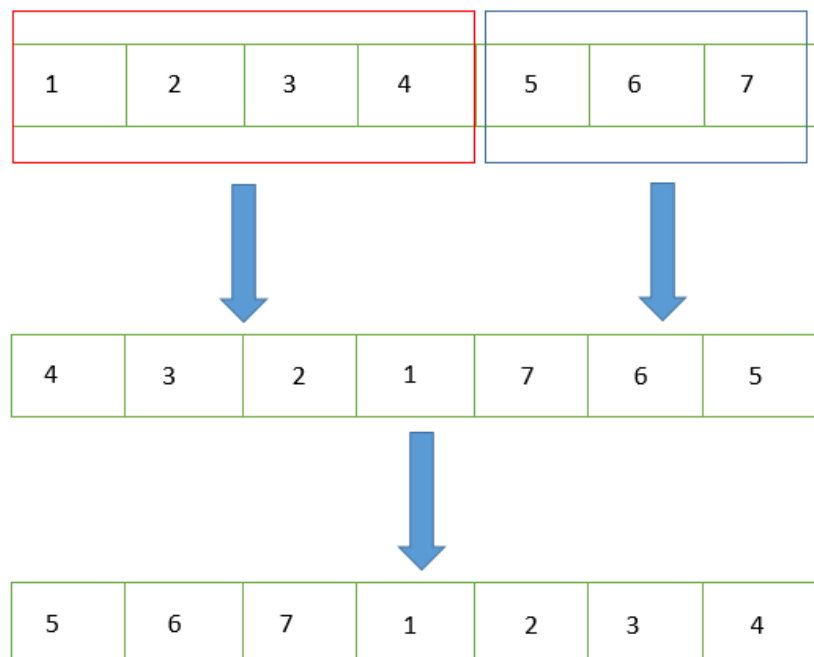
输入: $[-1, -100, 3, 99]$ 和 $k = 2$

输出: $[3, 99, -1, -100]$

解释:

向右旋转 1 步: $[99, -1, -100, 3]$

向右旋转 2 步: $[3, 99, -1, -100]$



```
void reverse(int* nums,int start,int end)
{
    while(start<end)
    {
        int tmp=nums[start];
        nums[start]=nums[end];
        nums[end]=tmp;
        start++;
        end--;
    }
}

void rotate(int* nums, int numsSize, int k){
    reverse(nums,0,numsSize-k%numsSize-1);
    reverse(nums,numsSize-k%numsSize,numsSize-1);
    reverse(nums,0,numsSize-1);
}
```