

Description

Solution

Discuss (999+)

Submissions

189. Rotate Array

Medium 7487 1157 Add to List Share

Given an array, rotate the array to the right by k steps, where k is non-negative.

Example 1:

Input: $nums = [1,2,3,4,5,6,7]$, $k = 3$

Output: $[5,6,7,1,2,3,4]$

Explanation:

rotate 1 steps to the right: $[7,1,2,3,4,5,6]$

rotate 2 steps to the right: $[6,7,1,2,3,4,5]$

rotate 3 steps to the right: $[5,6,7,1,2,3,4]$

Example 2:

Input: $nums = [-1,-100,3,99]$, $k = 2$

Output: $[3,99,-1,-100]$

Explanation:

rotate 1 steps to the right: $[99,-1,-100,3]$

rotate 2 steps to the right: $[3,99,-1,-100]$

Constraints:

- $1 \leq nums.length \leq 10^5$
- $-2^{31} \leq nums[i] \leq 2^{31} - 1$
- $0 \leq k \leq 10^5$

Follow up:

- Try to come up with as many solutions as you can. There are at least **three** different ways to solve this problem.
- Could you do it in-place with $O(1)$ extra space?

Java

Autocomplete

```
1 //Constant space solution using rotation
2 class Solution {
3
4     private void reverseArray(int[] array, int startIndex, int endIndex){
5         while(startIndex <= endIndex){
6             int temp = array[startIndex];
7
8             array[startIndex++] = array[endIndex];
9             array[endIndex--] = temp;
10        }
11    }
12
13    public void rotate(int[] nums, int k) {
14
15        k = k % nums.length;
16
17        reverseArray(nums, 0, nums.length - 1);
18
19        reverseArray(nums, 0, k-1);
20        reverseArray(nums, k, nums.length - 1);
21    }
22 }
```

Your previous code was restored from your local storage. [Reset to default](#)

Problems

Pick One

< Prev 355 Next >

Console ⁱ Contribute ⁱ

Run Code ^

Submit

https://leetcode.com/problems/rotate-array/

1/1