



19、一维数组循环左移

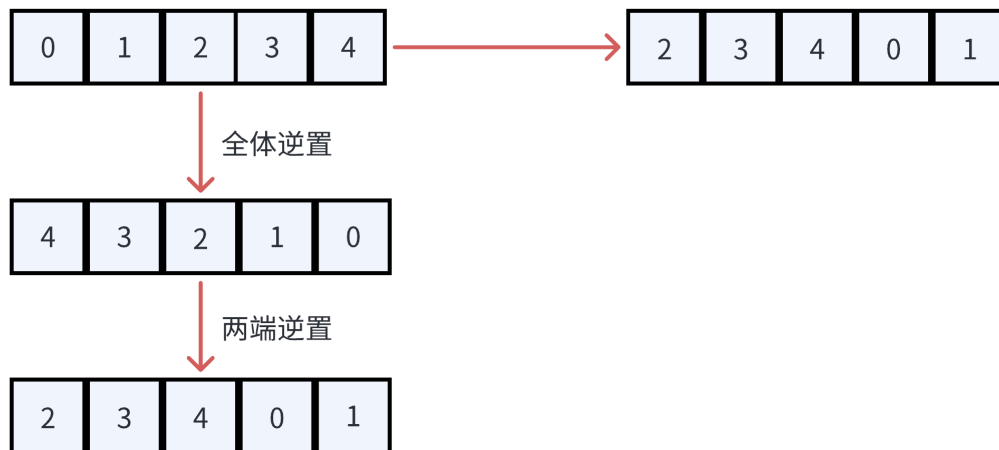
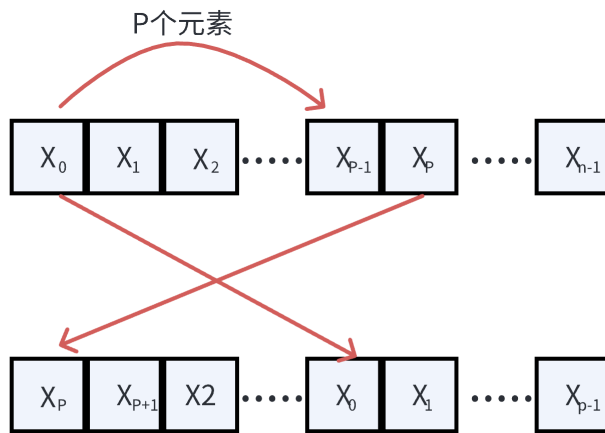
24计算机考研成员一战成硕！



题目描述：

【2010统考真题】设将 $n(n>1)$ 个整数存放于一维数组 R 中。设计一个在时间和空间两方面都**尽可能高效**的算法。将 R 中保存的序列循环左移 $p(0<p<n)$ 个位置，即将 R 中的数据由 $(x(0), x(1), x(2), \dots, x(n-1))$ 变换为 $(x(p), x(p+1), \dots, x(0), x(1), \dots, x(p-1))$ 要求：

- 1) 给出算法的基本设计思想。
- 2) 根据设计思想，采用C或C++或Java语言描述算法，关键之处给出注释。
- 3) 说明你所设计算法的时间复杂度和空间复杂度。



🕒 倒计时

1、知识点及难度



解人：多动症男孩

难度：中等偏下

知识点&注意点：

1. 除了本题这种最优解外，还可以通过“辅助数组”的方式解题，思路会比较清晰。

时间复杂度：翻转数组的操作只需要 $O(n)$ 的时间，其中 n 为数组的大小，因此整个算法的时间复杂度为 $O(n)$ 。

空间复杂度：仅常数级，没有额外的辅助空间

2、算法题

思路

1. 将数组分为两部分，前 p 个元素和剩余的 $n-p$ 个元素。
2. 分别翻转这两部分数组，得到两个翻转后的数组。
3. 然后再对整体进行一次翻转。

基本实现-C

```
1 // 翻转数组
2 void reverseArray(int arr[], int start, int end){
3     while(start < end) {
4         int temp = arr[start];
5         arr[start] = arr[end];
6         arr[end] = temp;
7         start++;
8         end--;
9     }
10 }
11
12 // 循环左移函数
13 void leftRotate(int arr[], int n, int p) {
14     if(p > n || p < 0) return;
15
16     // 翻转前p个元素
17     reverseArray(arr, 0, p - 1); // 这里是按照数组下标
18     // 翻转剩余的n-p个元素
19     reverseArray(arr, p, n - 1);
```

```
20 // 翻转整个数组
21 reverseArray(arr, 0, n - 1);
22 }
```

基本实现C++

```
1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
4
5 using namespace std;
6
7 // 翻转数组的函数
8 void reverseArray(vector<int>& arr, int start, int end) {
9     while (start < end) {
10         swap(arr[start], arr[end]);
11         start++;
12         end--;
13     }
14 }
15
16 // 循环左移函数
17 void leftRotate(vector<int>& arr, int p) {
18     int n = arr.size(); // 数组长度
19     if(p > n || p < 0) return;
20
21     // 翻转前p个元素
22     reverseArray(arr, 0, p - 1);
23     // 翻转剩余的n-p个元素
24     reverseArray(arr, p, n - 1);
25     // 翻转整个数组
26     reverseArray(arr, 0, n - 1);
27 }
28
```

3、总结



总结栏

蓝蓝B站首页：[蓝蓝希望你上岸呀B站首页](#)

蓝蓝公众号：[算法训练营9分计划](#)

蓝蓝知识星球介绍：[📖 关于知识星球的权益](#)

如何在星球打卡记录：z

- 计算机考研数据结构算法专项day[1/60]:
- 学习内容：最好能发出自己写的图片
- 遇到的问题：如果无就不用写了
- 小结：这部分一周写一次即可。