



## 21、求主元素

24计算机考研成员一战成硕！



题目描述：

【2013统考真题】已知一个整数序列 $A=(a_1, a_2, a_3, \dots, a_n)$ ，其中 $0 \leq a(i) < n$  ( $0 \leq i < n$ )。若存在 $a(p_1)=a(p_2)=\dots=a(p_m)=x$ 且 $m > n/2$  ( $0 \leq p(k) < n$ ,  $1 \leq k \leq m$ )，则称 $x$ 为 $A$ 的主元素。例如 $A=(0, 5, 5, 3, 5, 7, 5, 5)$ ，则5为主元素；又如 $Z=(0, 5, 5, 3, 5, 1, 5, 7)$ ，则 $A$ 中没有主元素。假设 $A$ 中的 $n$ 个元素保存在一个一维数组中，请设计一个**尽可能高效**的算法，找出 $A$ 的主元素。若存在主元素，则输出该元素；否则输出-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。

A数组	0	5	5	3	5	7	5	5
-----	---	---	---	---	---	---	---	---

  

辅助数组	0	0	0	0	0	0	0	0
	0	1	2	3	4	5	6	7

  

	1	0	0	1	0	5	0	1
	0	1	2	3	4	5	6	7

## 1、知识点及难度



**解人：多动症男孩**

**难度：简单**

**知识点：**

1. 投票算法是一种高效寻找主元素的方法，通过不断抵消不同元素来寻找候选主元素，从而减少了额外空间的使用。
2. 在找到候选主元素后，需要再次遍历数组统计其出现次数，以判断是否为真正的主元素。
3. 除此之外，本题也可以使用“辅助数组的方式”，思路会更加清晰！

**时间复杂度：** $O(n)$ ,只需要遍历数组两次，其中 $n$ 为数组的长度。

**空间复杂度：** $O(1)$ ,仅使用了常数级的额外空间，无其他辅助空间。

## 2、算法题

思路

1. 设置一个候选主元素candidate和计数器count，初始时分别为第一个元素和1。
2. 从第二个元素开始遍历数组，如果当前元素与候选元素相同，则计数器加1，否则计数器减1。

3. 每次当计数器值为0的时候，需要重新选择当前元素为候选元素，并将计数器设置为1。
4. 遍历完整个数组后，若计数器大于0，则得到主元素的候选，然后遍历数组统计元素在数组中出现的次数，判断是否满足主元素的条件。

## 基本实现-C

```
1 #include <stdio.h>
2
3 // 寻找主元素的函数
4 int findMajorityElement(int arrs[], int size) {
5     // 初始化候选主元素和计数器
6     int candidate = arrs[0];
7     int count = 1;
8
9     // 寻找候选主元素
10    for (int i = 1; i < size; i++) {
11        if(arrs[i] == candidate) {
12            count++;
13        }else {
14            count--;
15        }
16        if(count == 0) {
17            candidate = arrs[i];
18            count = 1;
19        }
20    }
21
22    // 统计候选主元素出现的次数，判断时候为主元素
23    count = 0; // 复用一下计数器
24    for(int i = 0; i < size; i++) {
25        if(arrs[i] == candidate) count++;
26    }
27
28    if(count > size / 2) return candidate;
29    else return -1;
30 }
```

## 基本实现C++

```
1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
```

```

4
5 using namespace std;
6
7 int findMajorityElement(vector<int>& arrs) {
8     // 初始化候选主元素和计数器
9     int candidate = arr[0];
10    int count = 1;
11
12    // 寻找候选主元素
13    for(auto arr: arrs) {
14        if(arr == candidate) count++;
15        else count--;
16        if(count == 0) {
17            candidate = arr;
18            count = 1;
19        }
20    }
21
22    // 统计候选主元素出现的次数，判断时候为主元素
23    count = 0; // 复用一下计数器
24    for(auto arr: arrs) {
25        if(arr == candidate) count++;
26    }
27    if(count > size / 2) return candidate;
28    else return -1;
29 }
30

```

### 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：**

- 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。