

| 02.2 - Majority

| 多数元素

数组中的多数元素是指在数组中出现次数超过 $n/2$ 的元素，其中 n 是数组的总元素数量

| 寻找数组中的多数元素

```
Majority(A[1,...,n]):  
    if |A| = 0  
        output "no"  
    else if |A| = 1  
        output A[1]  
    else 检查 A[n] 是否为多数元素 //处理奇数数组  
        if A[n] 是多数元素, 则  
            output A[n]  
        else  
            discard  
  
    初始化大小为 |A|/2 的数组 B  
    设置 j = 0  
    for i from 1 to n/2, do:  
        if A[2i-1] = A[2i], then  
            j = j + 1  
            B[j] = A[2i]  
  
    调用 Majority(B[1,...,j])  
    if Majority(B[1,...,j]) returns x, then  
        iterate array A 并计算 x 的出现次数  
        if x 的出现次数超过 n/2, then  
            output x  
        else  
            output "no"
```

| 算法正确性

Lemma: 如果 x 是 A 中的多数元素，那么 x 也是 B 中的多数元素

反证法:

假设 x 是 A 中的多数元素，但不是 B 中的多数元素

- m 为 x 在 A 中出现的次数
- k 为 x 在 B 中出现的次数

根据假设，其他元素在 B 中至少出现了 k 次
这意味着其他元素在 A 中至少出现了：

- $2k$ 次
- $m - 2k$ 次

有 $2k + (m - 2k) = m$, 这意味着 x 在 A 中不是多数元素, 与事实矛盾
因此 Lemma 正确

时间复杂度

运行时间的递推公式为:

$$T(n) \leq T\left(\frac{n}{2}\right) + cn$$

我们将证明:

$$T(n) \leq 2cn$$

归纳法 Inductive:

Base case: 当 $n = 1$ 时, $T(1) \leq c$

归纳假设: 假设 $T\left(\frac{n}{2}\right) \leq 2c\left(\frac{n}{2}\right)$

我们有 $T(n) \leq T\left(\frac{n}{2}\right) + cn$

根据归纳假设 $T(n) \leq 2c\left(\frac{n}{2}\right) + cn = 2cn$

$T(n) \leq 2cn$ 得证