

剑指39 数组中出现次数超过一半的数字

什么投票法???

方法1 哈希表统计法 时空间复杂度都 $O(n)$

方法2 数组排序法 将数组 `nums` 排序，数组中点的元素 一定为众数。

方法3 摩尔投票法 核心理念为 票数正负抵消。此方法时间和空间复杂度分别为 $O(N)$ 和 $O(1)$ ，为本题的最佳解法。

摩尔投票法中定义“数组中出现次数超过一半的数字”简称为“众数”。

推论1：如果记众数的票数为+1，非众数的票数为-1，则一定有所有数字票数和 >0

推论2：若数组的前 a 个数字的票数和 $=0$ ，则剩下的 $n-a$ 个数字的票数和仍 >0 ，即后 $n-a$ 个数字的众数仍为 x 。

```
1 // 我的哈希表统计法:
2 class Solution {
3 public:
4     int majorityElement(vector<int>& nums) {
5         int hlen = nums.size() / 2;
6         map<int, int> dic;
7         int res = nums[0];
8         for (int num : nums) {
9             if (dic.find(num) == dic.end()) {
10                 dic[num] = 1;
11             }
12             else {
13                 dic[num] = dic[num] + 1;
14                 if (dic[num] > hlen) {
15                     res = num;
16                     break;
17                 }
18             }
19         }
20         return res;
21     }
22 };
```

```
1 // k神摩尔投票法答案:
2 class Solution {
3 public:
4     int majorityElement(vector<int>& nums) {
5         int sum = 0;
6         int res = 0;
7         for (int i = 0; i < nums.size(); i++) {
8             if (sum == 0) {
9                 res = nums[i];
10            }
11            sum += nums[i] == res ? 1 : -1;
```

```
12     }  
13     return res;  
14 }  
15 };
```