

169. Majority Element

⌵ Difficulty	easy
☑	✓
📅 Finished	@July 12, 2023
⋮ Problem	array
⋮ Previously asked company	Facebook
⌵ website	leetcode

Question:

Given an array `nums` of size `n`, return *the majority element*.

The majority element is the element that appears more than `[n / 2]` times. You may assume that the majority element always exists in the array.

Example 1:

Input: `nums = [3,2,3]`
Output: `3`

Example 2:

Input: `nums = [2,2,1,1,1,2,2]`
Output: `2`

Optimal solution:

Time complexity: $O(n)$

Space complexity: $O(n)$

```
class Solution(object):
    def majorityElement(self, nums):
        dict = {}
        m = len(nums)//2
        for i in nums:
            dict[i] = 1 + dict.get(i,0)
            if dict[i] > m:
                return i
```

Follow-up: Could you solve the problem in linear time and in `O(1)` space?

Optimal solution:

Time complexity: $O(n)$

Space complexity: $O(1)$

```
class Solution(object):
    def majorityElement(self, nums):
        res, count = 0, 0

        for i in nums:
            if count == 0:
                res = i
            count += (1 if i == res else -1)

        return res
```