

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

## 169. Majority Element

[Easy](#) [Topics](#) [Lock](#)

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

The majority element is the element that appears more than  $\lfloor n / 2 \rfloor$  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**Constraints:**

- `n == nums.length`
- `1 <= n <= 5 * 104`

[22.4K](#) [504](#) [Star](#) [Share](#) [?](#)

Solved

&lt;/&gt; Code

Java Auto

```
1 class Solution {  
2     public int majorityElement(int[] nums) {  
3         int n = nums.length;  
4         int count = 0;  
5         int candidate = -1;  
6  
7         for (int num : nums) {  
8             if (count == 0) {  
9                 candidate = num;  
10                count = 1;  
11            }  
12            else if (num == candidate) {  
13                count++;  
14            }  
15        }  
16    }  
17}
```

Saved

 Testcase [Test Result](#)[Case 1](#) [Case 2](#) [+](#)

nums =

[3,2,3]

</> Source [?](#)