



Started on	Wednesday, 17 September 2025, 1:54 PM
State	Finished
Completed on	Wednesday, 17 September 2025, 2:22 PM
Time taken	28 mins 13 secs
Marks	1.00/1.00
Grade	10.00 out of 10.00 (100%)

**Question 1** | Correct Mark 1.00 out of 1.00

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`
- `-231 <= nums[i] <= 231 - 1`

**For example:**

Input	Result
3 3 2 3	3
7 2 2 1 1 1 2 2	2

**Answer:** (penalty regime: 0 %)

```

1  #include<stdio.h>
2  int majorityElements(int* nums,int n){
3      int count=0,candidate=0;
4      for(int i=0;i<n;i++){
5          if(count==0) candidate=nums[i];
6          if(nums[i]==candidate) count++;
7          else count--;
8      }
9      return candidate;
10 }
11 int main(){
12     int n;
13     scanf("%d",&n);
14     int nums[n];
15     for(int i=0;i<n;i++) scanf("%d",&nums[i]);
16     printf("%d\n", majorityElements(nums, n));
17     return 0;
18 }
```

	Input	Expected	Got	
✓	3	3	3	✓
	3 2 3			

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)