

MUKESH V A 2024-CSD-A ▾**M2****Started on** Thursday, 4 September 2025, 9:02 AM**State** Finished**Completed on** Thursday, 4 September 2025, 9:27 AM**Time taken** 25 mins 36 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
3 2 3	
7	2
2 2 1 1 1 2 2	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2 int main()
3 {
4     int n; scanf("%d", &n);
5     int a[n];
6     for (int i = 0; i < n; i++) scanf("%d", &a[i]);
7     int majority(int l, int r)
8     {
9         if (l == r) return a[l];
10        int m = (l + r) / 2;
11        int left = majority(l, m);
12        int right = majority(m + 1, r);
13        if (left == right) return left;
14
15        int cl = 0, cr = 0;
16        for (int i = l; i <= r; i++)
17        {
18            if (a[i] == left) cl++;
19            else if (a[i] == right) cr++;
20        }
21        return cl > cr ? left : right;
22    }
23    printf("%d\n", majority(0, n - 1));
24    return 0;
25 }
```

	Input	Expected	Got	
✓	3 3 2 3	3	3	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)