Started on	Tuesday, 1 October 2024, 1:43 PM
State	Finished
Completed on	Tuesday, 1 October 2024, 2:39 PM
Time taken	55 mins 57 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 `⌊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
```

Constraints:

- `n == nums.length`
- `1 <= n <= 5 * 10^4`
- `-2^31 <= nums[i] <= 2^31 - 1`

For example:

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

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int count(int arr[], int n, int n1){
    int c = 0;
    for(int i=0;i<n;i++){
        if(arr[i]==n){
            c++;
        }
    }
    return c;
}

int major(int arr[], int low, int high){
    if(low==high){
        return arr[0];
    }

    int mid = (low+high)/2;
    int left = major(arr, low, mid);
    int right = major(arr, mid+1, high);

    if(left == right){
        return left;
    }

    int lc = count(arr, high-low+1, left
    int rc = count(arr, high-low+1, righ
```

```
    if(lc>rc){
        return lc;
    }
    else{
        return rc;
    }
}

int main(){
    int n ;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    int ans = major(arr,0,n-1);
    printf("%d",ans);
}
```

	Input	Expected	Got	
✔	3 3 2 3	3	3	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 1-Number of Zeros in a Given Array

Jump to...

3-Finding Floor Value ►