Started on	Thursday, 12 September 2024, 10:06 AM
State	Finished
Completed on	Thursday, 12 September 2024, 11:00 AM
Time taken	53 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>
//using divide and canquer
int cc(int arr[],int l,int r,int c) {
    int count=0;
    for (int i=l;i<=r;i++){
        if (arr[i]==c) count++;
    }
    return count;
}
int fm(int arr[],int l,int r) {
    if (l>r) return -1;
    if (l==r) return arr[l];
    int m=l+(r-l)/2;
    int lm=fm(arr,l,m);
    int rm=fm(arr,m+1,r);
    if (lm==rm)
        return lm;
    int lc=cc(arr,l,r,lm);
    int rc=cc(arr,l,r,rm);
    return (lc>rc)?lm:rm;
}
int main() {
    int n;
    scanf("%d", &n);
    int arr[n];
    for (int i=0;i<n;i++) {
        scanf("%d",&arr[i]);
    }
    int maj=fm(arr,0,n-1);
    printf("%d\n",maj);
    return 0;
}

/*
without using divide and conquer
#include <stdio.h>
int maj(int* a,int n) {
```

```
    int count=0,c=0;
    for (int i=0;i<n;i++) {
        if(count==0)
            c=a[i];
        count+=(c==a[i])?1:-1;
    }
    return c;
}
int main() {
    int n;
    scanf("%d",&n);
    int a[n];
    for (int i=0;i<n;i++){
        scanf("%d", &a[i]);
    }
```

	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 ►