Dashboard / My courses / CS23331-DAA-2023-CSE / Divide and Conquer / 2-Majority Element

Started on	Tuesday, 8 October 2024, 1:52 PM
State	Finished
Completed on	Tuesday, 8 October 2024, 1:52 PM
Time taken	21 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>
#define MAX 1000
int findMostFrequent(int arr[], int n) {
    int freq[MAX] = {0};
    int maxCount = 0;
    int mostFrequent = arr[0];
    for (int i = 0; i < n; i++) {
        freq[arr[i]]++;
        if (freq[arr[i]] > maxCount) {
            maxCount = freq[arr[i]];
            mostFrequent = arr[i];
        }
    }
    return mostFrequent;
}

int main() {
    int n;
    scanf("%d", &n);
    int arr[n];
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }
    int result = findMostFrequent(arr, n);
    printf("%d\n", result);
    return 0;
}
```

	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 ►