Majority Element
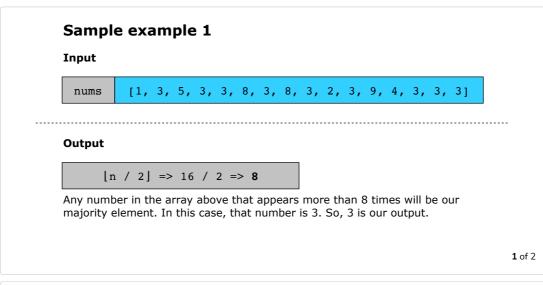
Try to solve the Majority Element problem.

Statement

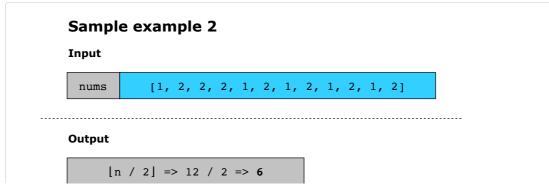Given an array, `nums` of size n, return the majority element.

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.

Constraints:

- $n ==$ `nums.length`
- $1 \leq n \leq 5 * 10^4$
- $-10^9 \leq$ `nums[i]` $\leq 10^9$

Examples

Sample example 1

Input

nums	[1, 3, 5, 3, 3, 8, 3, 8, 3, 2, 3, 9, 4, 3, 3, 3]

Output

$\lfloor n / 2 \rfloor$ => 16 / 2 => **8**

Any number in the array above that appears more than 8 times will be our majority element. In this case, that number is 3. So, 3 is our output.

Sample example 2

Input

nums	[1, 2, 2, 2, 1, 2, 1, 2, 1, 2, 1, 2]

Output

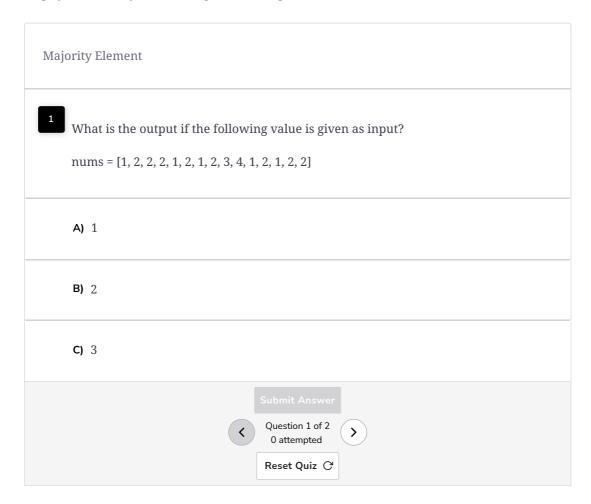$\lfloor n / 2 \rfloor$ => 12 / 2 => **6**

Any number in the array above appears more than 6 times will be our majority element. In this case, that number is 2. So, 2 is our output.

— ⟦⟧

Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

Majority Element

1

What is the output if the following value is given as input?

nums = [1, 2, 2, 2, 1, 2, 1, 2, 3, 4, 1, 2, 1, 2, 2]

A) 1

B) 2

C) 3

Submit Answer

< Question 1 of 2
0 attempted >

Reset Quiz ↻

Try it yourself

Implement your solution in the following coding playground:

☕ Java

usercode > FindMajorityElement.java

```
3    public static int findMajorityElement(int[] nums) {
4
5        // Write your code here
6
7        return 0;
8    }
9  }
```

Test Cases | Results

Case 1 | Case 2 | Case 3

Input #1

[1,2,1,1,1]

Majority Element

💡 Need a Hint?

← Back

Sum of Two Integers

Next →

Unique Paths

☑ Mark as Completed