Problem: Find the Majority Element

Company: MongoDB

Scenario

You are given a list of integers. A *majority element* in a list is defined as the element that appears more than ⌊n/2⌋ times, where *n* is the length of the list.
You can assume that such an element always exists.

Problem Statement

Given a list of elements, find the majority element, which appears more than half the time (>= floor(len(lst) / 2.0)).

You can assume that such element exists.

For example, given [1, 2, 1, 1, 3, 4, 0], return 1.

Input Format

- First line: An integer `n` (size of the list).
- Second line: `n` space-separated integers representing the elements of the list.

Output Format

- Print the majority element.

Example

Input

```
7
1 2 1 1 3 4 0
```

Output

```
1
```

Constraints

- $1 \leq n \leq 10^5$
- $-10^9 \leq A[i] \leq 10^9$

Hints

- Use a HashMap / dictionary to count frequencies, or
- Use the Boyer–Moore Majority Vote Algorithm for O(n) time and O(1) space.
  LeetCode link : ["Majority Element" – LeetCode 169]