Problem 136: Single Number

Problem Information

Difficulty: Easy
Acceptance Rate: 76.87%
Paid Only: No
Tags: Array, Bit Manipulation

Problem Description

Given a **non-empty** array of integers `nums`, every element appears _twice_ except for one. Find that single one.

You must implement a solution with a linear runtime complexity and use only constant extra space.

Example 1:

Input: nums = [2,2,1]

Output: 1

Example 2:

Input: nums = [4,1,2,1,2]

Output: 4

Example 3:

Input: nums = [1]

Output: 1

Constraints:

* `1 <= nums.length <= 3 * 104` * `-3 * 104 <= nums[i] <= 3 * 104` * Each element in the array appears twice except for one element which appears only once.

Code Snippets

C++:

```
class Solution {
public:
int singleNumber(vector<int>& nums) {

}
};
```

Java:

```
class Solution {
public int singleNumber(int[] nums) {

}
}
```

Python3:

```
class Solution:
def singleNumber(self, nums: List[int]) -> int:
```