Problem 3674: Minimum Operations to Equalize Array

Problem Information

Difficulty: Easy
Acceptance Rate: 56.44%
Paid Only: No
Tags: Array, Bit Manipulation, Brainteaser

Problem Description

You are given an integer array `nums` of length `n`.

In one operation, choose any subarray `nums[l...r]` (`0 <= l <= r < n`) and **replace** each element in that subarray with the **bitwise AND** of all elements.

Return the **minimum** number of operations required to make all elements of `nums` equal.

A **subarray** is a contiguous **non-empty** sequence of elements within an array.

Example 1:

Input: nums = [1,2]

Output: 1

Explanation:

Choose `nums[0...1]`: `(1 AND 2) = 0`, so the array becomes `[0, 0]` and all elements are equal in 1 operation.

Example 2:

Input: nums = [5,5,5]

Output: 0

Explanation:

`nums` is `[5, 5, 5]` which already has all elements equal, so 0 operations are required.

Constraints:

* `1 <= n == nums.length <= 100` * `1 <= nums[i] <= 105`

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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