Problem 3353: Minimum Total Operations

Problem Information

Difficulty: Easy
Acceptance Rate: 63.52%
Paid Only: Yes
Tags: Array

Problem Description

Given an array of integers `nums`, you can perform _any_ number of operations on this array.

In each **operation** , you can:

* Choose a **prefix** of the array. * Choose an integer `k` (which can be negative) and add `k` to each element in the chosen prefix.

A **prefix** of an array is a subarray that starts from the beginning of the array and extends to any point within it.

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

Example 1:

Input: nums = [1,4,2]

Output: 2

Explanation:

* **Operation 1** : Choose the prefix `[1, 4]` of length 2 and add -2 to each element of the prefix. The array becomes `[-1, 2, 2]`. * **Operation 2** : Choose the prefix `[-1]` of length 1 and add 3 to it. The array becomes `[2, 2, 2]`. * Thus, the minimum number of required operations is 2.

Example 2:

Input: nums = [10,10,10]

Output: 0

Explanation:

* All elements are already equal, so no operations are needed.

Constraints:

* `1 <= nums.length <= 105` * `-109 <= nums[i] <= 109`

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:
```