Problem 3192: Minimum Operations to Make Binary Array Elements Equal to One II

Problem Information

Difficulty: Medium
Acceptance Rate: 64.65%
Paid Only: No
Tags: Array, Dynamic Programming, Greedy

Problem Description

You are given a binary array `nums`.

You can do the following operation on the array **any** number of times (possibly zero):

* Choose **any** index `i` from the array and **flip** **all** the elements from index `i` to the end of the array.

Flipping an element means changing its value from 0 to 1, and from 1 to 0.

Return the **minimum** number of operations required to make all elements in `nums` equal to 1.

Example 1:

Input: nums = [0,1,1,0,1]

Output: 4

Explanation: We can do the following operations:

* Choose the index `i = 1`. The resulting array will be `nums = [0,_**0**_ ,_**0**_ ,_**1**_ ,_**0**_]`. * Choose the index `i = 0`. The resulting array will be `nums = [_**1**_ ,_**1**_ ,_**1**_ ,_**0**_ ,_**1**_]`. * Choose the index `i = 4`. The resulting array will be `nums = [1,1,1,0,_**0**_]`. * Choose the index `i = 3`. The resulting array will be `nums = [1,1,1,_**1**_

,_**1**_]`.

Example 2:

Input: nums = [1,0,0,0]

Output: 1

Explanation: We can do the following operation:

* Choose the index `i = 1`. The resulting array will be `nums = [1,_**1**_ ,_**1**_ ,_**1**_]`.

Constraints:

* `1 <= nums.length <= 105` * `0 <= nums[i] <= 1`

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:
```