Problem 1526: Minimum Number of Increments on Subarrays to Form a Target Array

Problem Information

Difficulty: Hard
Acceptance Rate: 78.02%
Paid Only: No
Tags: Array, Dynamic Programming, Stack, Greedy, Monotonic Stack

Problem Description

You are given an integer array `target`. You have an integer array `initial` of the same size as `target` with all elements initially zeros.

In one operation you can choose **any** subarray from `initial` and increment each value by one.

Return _the minimum number of operations to form a_`target` _array from_`initial`.

The test cases are generated so that the answer fits in a 32-bit integer.

Example 1:

Input: target = [1,2,3,2,1] **Output:** 3 **Explanation:** We need at least 3 operations to form the target array from the initial array. [**_0,0,0,0,0_**] increment 1 from index 0 to 4 (inclusive). [1,**_1,1,1_** ,1] increment 1 from index 1 to 3 (inclusive). [1,2,**_2_** ,2,1] increment 1 at index 2. [1,2,3,2,1] target array is formed.

Example 2:

Input: target = [3,1,1,2] **Output:** 4 **Explanation:** [**_0,0,0,0_**] -> [1,1,1,**_1_**] -> [**_1_** ,1,1,2] -> [**_2_** ,1,1,2] -> [3,1,1,2]

Example 3:

Input: target = [3,1,5,4,2] **Output:** 7 **Explanation:** [**_0,0,0,0,0_**] -> [**_1_** ,1,1,1,1] -> [**_2_** ,1,1,1,1] -> [3,1,**_1,1,1_**] -> [3,1,**_2,2_** ,2] -> [3,1,**_3,3_** ,2] -> [3,1,**_4_** ,4,2] -> [3,1,5,4,2].

Constraints:

* `1 <= target.length <= 105` * `1 <= target[i] <= 105` * ■■■■■■■The input is generated such that the answer fits inside a 32 bit integer.

Code Snippets

C++:

```
class Solution {
public:
int minNumberOperations(vector<int>& target) {

}
};
```

Java:

```
class Solution {
public int minNumberOperations(int[] target) {

}
}
```

Python3:

```
class Solution:
def minNumberOperations(self, target: List[int]) -> int:
```