

Problem 2355: Maximum Number of Books You Can Take

Problem Information

Difficulty: Hard

Acceptance Rate: 39.37%

Paid Only: Yes

Tags: Array, Dynamic Programming, Stack, Monotonic Stack

Problem Description

You are given a **0-indexed** integer array `books` of length `n` where `books[i]` denotes the number of books on the `i`th shelf of a bookshelf.

You are going to take books from a **contiguous** section of the bookshelf spanning from `l` to `r` where `0 ≤ l ≤ r < n`. For each index `i` in the range `l ≤ i < r`, you must take **strictly fewer** books from shelf `i` than shelf `i + 1`.

Return the maximum number of books you can take from the bookshelf.

Example 1:

Input: `books = [8,5,2,7,9]` **Output:** 19 **Explanation:** - Take 1 book from shelf 1. - Take 2 books from shelf 2. - Take 7 books from shelf 3. - Take 9 books from shelf 4. You have taken 19 books, so return 19. It can be proven that 19 is the maximum number of books you can take.

Example 2:

Input: `books = [7,0,3,4,5]` **Output:** 12 **Explanation:** - Take 3 books from shelf 2. - Take 4 books from shelf 3. - Take 5 books from shelf 4. You have taken 12 books so return 12. It can be proven that 12 is the maximum number of books you can take.

Example 3:

****Input:**** books = [8,2,3,7,3,4,0,1,4,3] ****Output:**** 13 ****Explanation:**** - Take 1 book from shelf 0. - Take 2 books from shelf 1. - Take 3 books from shelf 2. - Take 7 books from shelf 3. You have taken 13 books so return 13. It can be proven that 13 is the maximum number of books you can take.

****Constraints:****

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

Code Snippets

C++:

```
class Solution {
public:
    long long maximumBooks(vector<int>& books) {

    }
};
```

Java:

```
class Solution {
    public long maximumBooks(int[] books) {

    }
}
```

Python3:

```
class Solution:
    def maximumBooks(self, books: List[int]) -> int:
```