Problem 1413: Minimum Value to Get Positive Step by Step Sum

Problem Information

Difficulty: Easy
Acceptance Rate: 64.52%
Paid Only: No
Tags: Array, Prefix Sum

Problem Description

Given an array of integers `nums`, you start with an initial **positive** value _startValue_ _._

In each iteration, you calculate the step by step sum of _startValue_ plus elements in `nums` (from left to right).

Return the minimum **positive** value of _startValue_ such that the step by step sum is never less than 1.

Example 1:

Input: nums = [-3,2,-3,4,2] **Output:** 5 **Explanation:** If you choose startValue = 4, in the third iteration your step by step sum is less than 1. **step by step sum** **startValue = 4 | startValue = 5 | nums** (4 **-3**) = 1 | (5 **-3**) = 2 | -3 (1 **+2**) = 3 | (2 **+2**) = 4 | 2 (3 **-3**) = 0 | (4 **-3**) = 1 | -3 (0 **+4**) = 4 | (1 **+4**) = 5 | 4 (4 **+2**) = 6 | (5 **+2**) = 7 | 2

Example 2:

Input: nums = [1,2] **Output:** 1 **Explanation:** Minimum start value should be positive.

Example 3:

Input: nums = [1,-2,-3] **Output:** 5

Constraints:

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

Code Snippets

C++:

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

    }
};
```

Java:

```java
class Solution {
    public int minStartValue(int[] nums) {

    }
}
```

Python3:

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