Task Name

Peak Value Control 1

Difficulty

Medium

Problem Description

Move value only to the left any number of times. Find the smallest possible maximum value in the final array.

Problem Statement

You are given an array `a` of `n` non-negative integers.

In one operation, you may choose an index `i` such that `1 <= i < n` and `a[i] > 0`, then: - decrease `a[i]` by `1` - increase `a[i - 1]` by `1`

You may perform this operation any number of times.

Your task is to find the minimum possible value of the maximum element of the array after all operations.

Input Format

- The first line contains a single integer `n`.
- The second line contains `n` space-separated integers `a[0], a[1], ..., a[n - 1]`.

Constraints

- `2 <= n <= 10^5`
- `0 <= a[i] <= 10^9`

Output Format

Print one integer: the minimum possible value of the maximum element after performing any number of operations.

Example 1

Input

5 0 8 2 0 6

Output

4

Explanation

One optimal strategy is to move units leftward until the array becomes: `[4, 4, 4, 2, 2]`

The maximum value is `4`, and it cannot be made smaller.

Example 2

Input

6 1 0 10 0 0 1

Output

4

Explanation

The large value at index `2` can be spread to the left. A final configuration with maximum `4` is possible, so the answer is `4`.

Tags

Arrays, Greedy, Prefix Sums

Reference Solution Idea

For every prefix `a[0..i]`, its total sum cannot move to the right, so those `i + 1` positions must hold that sum. Therefore, the answer must be at least: `ceil(prefixSum / (i + 1))`

Take the maximum of this value over all prefixes.

Complexity

- Time: `O(n)`
- Space: `O(1)` extra space

Suggested testcase roles

- `input00/output00` -> Sample, strength `0`
- `input01/output01` -> Sample, strength `0`
- `input02/output02` -> Hidden, strength `50`
- `input03/output03` -> Hidden, strength `50`