Minimum Absolute Slice Problem Description

A non-empty zero-indexed array A of N integers is given. A pair of integers (P, Q), such that 0 P Q < N, is called a slice of array A. The sum of a slice (P, Q) is the total of A[P] + A[P+1] + ... + A[Q]. A min abs slice is a slice whose absolute sum is minimal.

For example, array A such that:

- A[0] = 2
- A[1] = -4
- A[2] = 6
- A[3] = -3
- A[4] = 9

contains the following slices, among others:

- (0, 1), whose absolute sum = |2 + (4)| = 2
- (0, 2), whose absolute sum = |2 + (4) + 6| = 4
- (0, 3), whose absolute sum = |2 + (4) + 6 + (3)| = 1
- (1, 3), whose absolute sum = |(4) + 6 + (3)| = 1
- (1, 4), whose absolute sum = |(4) + 6 + (3) + 9| = 8
- (4, 4), whose absolute sum = |9| = 9

Both slices (0, 3) and (1, 3) are min abs slices and their absolute sum equals 1.

Write a function:

int solution(int A[], int N);

that, given a non-empty zero-indexed array A consisting of N integers, returns the absolute sum of min abs slice.

For example, given:

- A[0] = 2
- A[1] = -4
- A[2] = 6
- A[3] = -3
- A[4] = 9

the function should return 1, as explained above.

Assume that:

•*N is an integer within the range [1..100,000];*

•*each element of array A is an integer within the range [10,000..10,000].*

Complexity:

•*expected worst-case time complexity is O(N*log(N));*

•*expected worst-case space complexity is O(N), beyond input storage (not counting the storage required for input arguments).*

Elements of input arrays can be modified.