

## Problem

You are given a binary array  $A$  of  $N$  elements. The array consists of 0's and 1's. You can perform the following operations as many times as possible:

- Select a subarray starting from the first index that is inversion-free and delete it.

Determine the minimum number of operations to delete the entire array.

- **Inversion free:** There are no two indices  $i$  and  $j$  in array  $A$  such that  $(i < j)$  and  $(A_i > A_j)$ .
- **Subarray:** A subarray is an array obtained after deleting some elements from the beginning (prefix) possibly 0 and deleting some elements from the end (suffix) possibly 0.

## Input format

- The first line contains an integer  $T$  denoting the number of test cases.
- The first line of each test case contains an integer  $N$  denoting the number of elements in array  $A$ .
- The second line contains  $N$  space-separated integers of array  $A$ .

## Output Format

Print  $T$  lines and for each test case:

- Print the minimum number of operations to delete the entire array.

## Constraints

$$1 \leq T \leq 20000$$

$$1 \leq N \leq 200000$$

$$0 \leq A_i \leq 1$$

Sample Input	Sample Output
3 4 0 0 1 1 2 1 0 2 0 0	1 2 1

Time Limit: 1

Memory Limit: 256

Source Limit: