Problem Bricks

C header	`bricks.h`
C++ header	`bricks.h`

Little Square and Little Triangle have recently received a very nice present from Little Circle.

The present consists of a sequence of N toy bricks of **pairwise different heights** numbered from 0 to $N-1$, which are arranged in a line. Each of the N bricks is either red or purple. The i^{th} brick, from left to right, has height $H[i]$ and colour code $C[i]$, where $C[i]$ can be either *true* (representing a red brick) or *false* (representing a purple brick).

After receiving the gift, Little Square and Little Triangle are eager to play with it. They start by counting how many *interesting* bricks there are. They consider a brick to be *interesting* if there is no higher brick of the same colour on its right.

However, they finish counting the *interesting* bricks in the original sequence very soon and are now looking for another challenge. They are wondering what is the maximum possible number of *interesting* bricks they could get if they modify the colour of **at most** one brick (from purple to red or from red to purple). Could you help them with this challenge?

Interaction protocol

The contestant must implement one function:

```
int solve(int N, bool C[], int H[]);
```

This function will be called **exactly once**, at the beginning of the interaction. Note that C and H do not necessarily have size N. They could have a larger size, in which case $H[i] = 0$ and $C[i] = false$ for all $i \geq N$.

The sample grader reads, on the first line, the integer N. On the second line it reads the values of C (with 0 representing *false* and 1 representing *true*), and on the third line it reads the array H. It outputs the result of calling `solve` on these arguments.

Attention! The contestant should not implement the main function.

Constraints

- $1 \leq N \leq 6 \cdot 10^6$
- $1 \leq H[i] \leq 2 \cdot 10^9$
- All heights are pairwise distinct.

Subtask 1 (9 points)

- No changes have to be made to the initial array.

Subtask 2 (21 points)

- $C[i] = true$ for all $0 \leq i \leq N-1$

Subtask 3 (13 points)

- $1 \leq N \leq 1000$

Subtask 4 (29 points)

- $1 \leq N \leq 200.000$

Subtask 5 (28 points)

- No additional constraints.

Example

input	output
6 0 0 1 0 0 1 7 8 6 2 3 5	5

Explanation

The answer is obtained by changing the colour of the first or second brick. After this change, 7, 8, 6, 3 and 5 are interesting.