Beats

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

Bartek is a well-known techno music producer, and he is currently perfecting his next hit. He has created a track consisting of n beats, numbered consecutively from 1 to n. After spending hours in a music editing program, he was about to export the mp3 file, but he noticed that the beats are currently arranged in the order $a_1, a_2, \ldots, a_n$. Bartek needs to obtain an increasing order of beats, that is, $1, 2, \ldots, n$.

The program allows two types of operations:

- Move the last beat to any position, e.g. $(2, 4, 1, 5, \underline{3}) \to (2, \underline{3}, 4, 1, 5)$.
- Move any beat to the beginning, e.g. $(2, 4, 1, \underline{5}, 3) \to (\underline{5}, 2, 4, 1, 3)$.

What is the minimum number of operations needed to arrange the beats in increasing order?

Input

The first line contains an integer n ($1 \leq n \leq 200\,000$), denoting the number of beats.

The second line contains n pairwise distinct integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq n$; $a_i \neq a_j$), representing the current order of beats in the program.

Output

Output a single integer – the minimum number of operations required.

Examples

standard input	standard output
6 3 2 5 4 1 6	4
9 3 1 2 5 6 4 9 7 8	6

Note

We start with the sequence of beats $(3, 2, 5, 4, 1, 6)$. Here is one way to achieve the desired order in four operations:

- Move the last beat two places to the left: $(3, 2, 5, \underline{6}, 4, 1)$.
- Move beat 2 to the beginning: $(\underline{2}, 3, 5, 6, 4, 1)$.
- Move beat 1 to the beginning: $(\underline{1}, 2, 3, 5, 6, 4)$.
- Move the last beat two places to the left: $(1, 2, 3, \underline{4}, 5, 6)$.

It is possible to obtain an increasing order in other ways using four operations, but it cannot be done in three operations, so the result is 4.