CCC '15 S3 - Gates

Time Limit: 1.0s **Memory Limit:** 256M
Python 2: 2.5s
Python 3: 2.5s

Canadian Computing Competition: 2015 Stage 1, Senior #3

For your birthday, you were given an airport.

The airport has G gates, numbered from 1 to G. P planes arrive at the airport, one after another. You are to assign the i^{th} plane to permanently dock at any gate $1, \ldots, g_i$ ($1 \leq g_i \leq G$), at which no previous plane has docked. As soon as a plane cannot dock at any gate, the airport is shut down and no future planes are allowed to arrive.

In order to keep the person who gave you the airport happy, you would like to maximize the number of planes starting from the beginning that can all dock at different gates.

Input Specification

The first line of input contains G ($1 \leq G \leq 10^5$), the number of gates at the airport.

The second line of input contains P ($1 \leq P \leq 10^5$), the number of planes which will land.

The next P lines contain one integer g_i, ($1 \leq g_i \leq G$), such that the i^{th} plane must dock at some gate from 1 to g_i, inclusive.

Note that for at least 20% of the marks for this question, $P \leq 2\,000$ and $G \leq 2\,000$.

Output Specification

Output the maximum number of planes that can land starting from the beginning.

Sample Input 1

```
4
3
4
1
1
```

Output for Sample Input 1

```
2
```

Explanation of Output for Sample Input 1

The first plane can go anywhere, but it is best to not put it into Gate 1. Notice that planes 2 and 3 both want to dock into Gate 1, so plane 3 is unable to dock.

Sample Input 2

```
4
6
2
2
3
3
4
4
```

Output for Sample Input 2

```
3
```

Explanation of Output for Sample Input 2

The first two planes will dock in gates 1 and 2 (in any order). The third plane must dock at Gate 3. Thus, the fourth plane cannot dock anywhere, and the airport is closed, even though plane 5 would have been able to dock.