

## Police Investigation (police)

Dark days here in Gotham: criminality is at its highest, and the fearsome William is still free. The police is non-stop searching for him, and the last clues point towards the long and creepy Terror Street.



Figure 1: Terror Street.

Terror Street is composed of  $N$  houses, numbered from 1 to  $N$ . It is conjectured that William hides in one of them... which one though?

The police starts searching from the first, and stops only when William is found. The agents, if they don't find the fugitive in a house, will interrogate the inhabitant until he gives them the number of the next house to check. So the agents go, house by house, following the information told by the citizens, until William is found, hoping he'll stay still!

Plot twist! You are William! And you don't like to get caught.

Knowing in which house you are, and what the citizens will tell to the police, count after how many wrong attempts you'll be found by the police... so that you can plan your last-minute escape accordingly!

👉 Among the attachments of this task you may find a template file `police.*` with a sample incomplete implementation.

### Input

The first line contains the integers  $N$  and  $K$ , the total number of houses and the number of William's house. The second line contains  $N$  integers  $V_i$ , the answer of the citizen in the  $i$ -th house.

### Output






You need to write a single line with an integer: the number of failed attempts of the police (zero if William is found in the first house, a positive number  $x$  if he's found after  $x$  attempts,  $-1$  if he cannot be found by the police).

## Constraints

- $1 \leq N, K \leq 100\,000$ .
- $1 \leq V_i \leq N$  for each  $i = 1 \dots N$ .

## Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

- **Subtask 1** (0 points)      Examples.  

- **Subtask 2** (20 points)       $N \leq 10$ .  

- **Subtask 3** (20 points)       $N \leq 1000$ .  

- **Subtask 4** (40 points)      William is always locatable by the police.  

- **Subtask 5** (20 points)      No additional limitations.  


## Examples

input	output
5 3 2 4 1 3 3	3
5 1 3 2 1 2 5	0
5 5 2 3 1 1 5	-1

## Explanation

In the **first sample case** the police starts from 1, then goes to 2, then to 4, and finally finds William at 3. It took 3 attempts before finding him.

In the **second sample case** William is in the first house, so there are no failing attempts.

In the **third sample case** the police starts from 1, then goes to 2, then to 3, and then back to 1, looping forever: William is safe and doesn't have to escape at all.