

## Problem F. Movies

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 256 mebibytes

Watching movies is a complicated affair. You have partitioned your available movies into two lists:

- the main list containing  $N$  movies,
- the secondary list containing  $K$  movies.

You therefore have a total of  $N + K$  movies. Each movie has a distinct integer rating, a value from the set  $\{1, 2, \dots, N + K\}$ .

Because you don't want to watch too many bad movies in a row or too many good movies in a row, you have devised the following algorithm for watching them:

- At each step, you take a movie from the main list and watch it. It then disappears from the main list.
- The movies that you select must always alternate between the worst available movie and the best available movie. The first selection must be the best movie. The movie with the lowest rating is considered to be the worst. The movie with the highest rating is considered to be the best. A movie is available only if it is contained in the main list.
- After watching the selected movie, you will choose another movie from the secondary list and insert it into the main list at any position. Once the secondary list is empty, you will stop watching movies. Note that this means that the main list will always have exactly  $N$  movies.

You want your main list to become sorted in ascending order according to movie rating. Because you are too lazy to do this properly, you won't do anything to the list itself. Instead, at each step of the algorithm, you will decide which movie to insert into the main list and where to insert it, so that the number of steps after which the main list becomes sorted is minimized.

### Input

The first line of input contains two integers  $N$  and  $K$  ( $1 \leq N \leq 100\,000$ ,  $1 \leq K \leq 200\,000$ ).

The second line contains  $N$  integers: the main list of movies.

The third line contains  $K$  integers: the secondary list of movies.

It is guaranteed that the union of the main list ratings and the secondary list ratings equals to the set  $\{1, 2, \dots, N + K\}$ .

### Output

Print a single line containing a single integer: the minimum number of steps needed to sort the main list, or the value  $-1$  if it is not possible.

### Example

standard input	standard output
5 5 3 1 5 2 4 6 8 7 9 10	4

## Explanation

During the first step, you will watch the movie with rating 5, because it is the best movie available. Then, you must replenish the main list. You do this by inserting the movie with rating 7 from the secondary list at the back of the main list. They now look like this:

$$\begin{aligned}\text{Main} &= [3\ 1\ 2\ 4\ 7] \\ \text{Secondary} &= [6\ 8\ 9\ 10]\end{aligned}$$

During the second step, it is time you watch the worst movie, so you watch the movie with rating 1. You replace it with the movie with rating 8, which you again place at the end of the list.

$$\begin{aligned}\text{Main} &= [3\ 2\ 4\ 7\ 8] \\ \text{Secondary} &= [6\ 9\ 10]\end{aligned}$$

During the third step, you watch the movie with rating 8 (best available), and add the movie with rating 9 at the end of the list.

$$\begin{aligned}\text{Main} &= [3\ 2\ 4\ 7\ 9] \\ \text{Secondary} &= [6\ 10]\end{aligned}$$

During the fourth step, you watch the movie with rating 2 (worst available) and add the movie with rating 10 at the end of the list.

$$\begin{aligned}\text{Main} &= [3\ 4\ 7\ 9\ 10] \\ \text{Secondary} &= [6]\end{aligned}$$

The main list is now sorted in ascending order, so the answer is 4. Note that, although all insertions in this example occurred at the end of the main list, this is not generally necessary. Movies may be inserted anywhere in the main list.