

Problems

Problem	JOKER	GALERIJA	VOKI
Source file	joker.pas joker.c joker.cpp	galerija.pas galerija.c galerija.cpp	voki.pas voki.c voki.cpp
Input file	joker.in	galerija.in	voki.in
Output file	joker.out	galerija.out	voki.out
Time limit per test	5 seconds	5 seconds	5 seconds
Number of tests	10	10	10
Points per test	5	6	9
Total points	50	60	90
	200		

JOKER

Mirko plays very interesting card game. This game is played with N cards one of which is a joker.

Mirko puts cards on the table **from left to right** so that the **joker is the rightmost card** on the table and then in determined number of moves removes the cards from the table.

He chooses two cards, card A and card B (**none of which is joker**) and then removes **all** cards **between** card A and card B , **including** A and B . He writes down that move on the paper as (A,B) , and then moves card from the right of the space left so that they maintain order.

After a certain number of moves, there is only one card left on the table, the joker, after which Slavko steals Mirko's piece of paper containing the moves made, and copies them to another piece of paper, in **a random order**. On occasions he will also **modify the numbers** themselves.

Write a program that will determine **some order (if it exists) of the moves** written on paper, so that the execution of the moves in that order will leave joker the only card on table.

Input data

In the first line of input file there are two integers N and K , separated by space, $1 \leq N \leq 1,000,000,000$, number of cards and number of moves

In every of the following K lines there are two integers A and B separated by space, $A \leq B$. They represent move (A, B) .

Output data

Output file should contain K lines. In i -th line should be i -th move from the text.

If ordering doesn't exist you should write **'-1'**.

Examples

joker.in

3 1
1 3

joker.out

-1

joker.in

5 2
2 3
1 2

joker.out

2 3
1 2

joker.in

9 3
1 3
4 4
3 6

joker.out

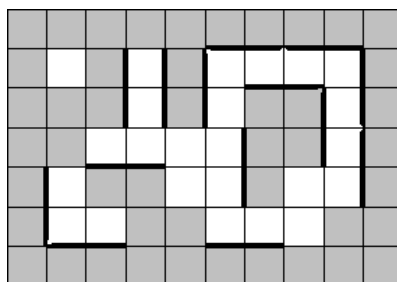
3 6
4 4
1 3

GALERIJA

Map of an art gallery can be represented by a square grid of M columns and N rows. Some of the squares are **walls** while others represents **empty space**.

Rooms consist of **adjacent free** squares (squares that share edge). On the walls of the rooms we want to hang as many pictures as possible. Every picture occupies exactly **two lengths** of a square and their height doesn't matter. In other words you can hang a picture on the wall if there are two consecutive walls and two free squares opposite of them (that two free squares should, of course, be at the same side of walls).

Write a program that computes the **maximum** number of pictures that can be hanged on the walls.



This picture shows one of the possible solutions for a gallery map.

Input data

In the first line of the input file there are two integers M and N , $1 \leq M, N \leq 1000$, number of the rows and number of the columns of the gallery.

In the next M rows there are N characters. Every character represents either a free square or wall. 'X' represents wall and '.' represents free space.

First and last row, as well as first and last column are walls.

Output data

In the first line of file you should write maximum number of pictures that can be hanged on walls of gallery.

Examples

galerija.in

```
3 3
XXX
X.X
XXX
```

galerija.out

0

galerija.in

```
5 5
XXXXX
X...X
X.XXX
X.X.X
XXXXX
```

galerija.out

4

galerija.in

```
7 10
XXXXXXXXXX
X.X.X...X
XXX.X.XX.X
XX...XX.X
X.XX..X..X
X..XX...XX
XXXXXXXXXX
```

galerija.out

14

VOKI

Substring of some string A is defined as one or more (**not necessary succeeding**) elements of the string with maintaining the sequence.

There are given two strings, string VOKI and string TOKI.

Write the program that will calculate the length of any **shortest** substring of string VOKI such as it **is not** substring of string TOKI.

Input data

In first line of input file there is string VOKI and in second one is string TOKI.

The only characters that will occur are lowercase characters of English alphabet ('a' - 'z'). String lengths will be less or equal to 1000.

Note: input data will be such so there will always be a solution.

Output data

In the first line of file you should print the **length** of wanted substring.

Examples

voki.in

ababaa

abbaa

voki.out

3

(eg. bab)

voki.in

babab

babba

voki.out

3

(eg. aab)

voki.in

banana

anbnaanbaan

voki.out

5

(eg. banna)