

---

**F010B. Threatening bishops****P67340\_en**

---

Consider a rectangular chessboard where there are some bishops. A bishop threatens to other one if both are in the same diagonal and there is not any other bishop between them. We want to know all the threats between bishops.

For instance, consider the following chessboard with 4 rows and 5 columns:

1				X	
2			X		
3				X	X
4	X				
	1	2	3	4	5

The bishop of the position (3,5) has not any threat. The bishop of the position (4,1) threatens the bishop of the position (2,3). The bishop of the position (2,3) threatens to the bishops of the positions (4,1), (1,4) and (3,4). The set of all the threats is:

$$\begin{aligned}(4,1) &\leftrightarrow (2,3) \\(2,3) &\leftrightarrow (4,1) \\(2,3) &\leftrightarrow (1,4) \\(1,4) &\leftrightarrow (2,3) \\(2,3) &\leftrightarrow (3,4) \\(3,4) &\leftrightarrow (2,3)\end{aligned}$$

Your task is to write a program that reads a chessboard with bishops and writes all their threats.

**Input**

The input starts with the number of rows and the number of columns of the chessboard (two integers strictly positive). Afterwards, there is a chessboard of the indicated size, with characters 'X' to indicate the bishops and dots to indicate the empty boxes.

**Output**

For each threat of a bishop to the position  $(r_1, c_1)$  to other bishop in the position  $(r_2, c_2)$ , print a row with the text " $(r_1, c_1) \leftrightarrow (r_2, c_2)$ ". The positions are numbered from top to bottom and from left to right, starting for 1.

**Observation**

The order which you write the lines is irrelevant. The Judge will consider as good any output that contains all the correct threats, with independence of their order.

### Sample input 1

```
4 5
...X.
..X..
...XX
X....
```

### Sample input 2

```
3 3
XXX
XXX
XXX
```

### Sample input 3

```
6 11
.....X.....
.....
.....
.....
.....
.....
XXXXXXXXXXXX
```

### Sample input 4

```
1 1
.
```

### Sample output 1

```
(1, 4) <-> (2, 3)
(2, 3) <-> (3, 4)
(2, 3) <-> (4, 1)
(2, 3) <-> (1, 4)
(3, 4) <-> (2, 3)
(4, 1) <-> (2, 3)
```

### Sample output 2

```
(1, 1) <-> (2, 2)
(1, 2) <-> (2, 3)
(1, 2) <-> (2, 1)
(1, 3) <-> (2, 2)
(2, 1) <-> (3, 2)
(2, 1) <-> (1, 2)
(2, 2) <-> (3, 3)
(2, 2) <-> (3, 1)
(2, 2) <-> (1, 3)
(2, 2) <-> (1, 1)
(2, 3) <-> (3, 2)
(2, 3) <-> (1, 2)
(3, 1) <-> (2, 2)
(3, 2) <-> (2, 3)
(3, 2) <-> (2, 1)
(3, 3) <-> (2, 2)
```

### Sample output 3

```
(1, 6) <-> (6, 11)
(1, 6) <-> (6, 1)
(6, 1) <-> (1, 6)
(6, 11) <-> (1, 6)
```

### Sample output 4

## Problem information

Author : Professorat de P1  
Translator : Carlos Molina  
Generation : 2024-05-02 21:44:38

© *Jutge.org*, 2006–2024.  
<https://jutge.org>