Knights and knaves

There is a row with n people, of which m are *knights* and the rest are *knaves*. Knights always tell the truth and knaves always lie. Initially you don't know which people are knights.

You can choose one of the persons and ask the following question: "In which direction is the nearest knight?". If the person you are asking is a knight, it will answer the direction in which there is the other (different) knight which is nearest (according to the position in the row). But if you ask a knave, it will answer the **opposite** direction to which the nearest knight is. It is guaranteed that the distribution of knights and knaves in the line will be such that the answers are uniquely determined, that is, there will not be any knight or knave such that the two nearest knights are found in the same direction.

Your goal is to identify the m knights doing at most $100 \cdot (m + 1)$ questions.

Input and output

This is an interactive problem. You must flush the output (`cout << endl` or `cout << flush` in C++, `System.out.flush()` in Java, `stdout.flush()` in Python).

The first line of input contains two integers n and m, the number of people and the number of knights. You must read these values before making any query.

For making a query you must write one line with the format `? i` , where i is the index of the person you ask, $0 \le i < n$. After that, you must read one character from the input, the result. The character will be `<` if the direction is to the left (that is, towards lower positions) or `>` if the direction is towards higher positions. If you make a invalid query or exceed the query limit, the result you read will be `-`, and your program should terminate immediately.

Once you have found the m knights, you must write a line with the format `! i_1 i_2 ... i_m` , where $i_1 < i_2 < \ldots < i_m$ are the positions of the knights **in ascending order**. After writing this, your program must terminate.

Sample

Input:

```
10 3
<
>
<
>
>
<
<
<
>
>
```

Output:

```
? 0
? 1
```

```
? 2
? 3
? 4
? 5
? 6
? 7
? 8
? 9
! 1 2 7
```

Explaination: Knights are in positions $1, 2, 7$ (0-indexed). First person in position 0 is asked: it has the nearest knight, 1, in direction `>`, but it is a knave, so it answers `<`. After that person 1 is asked, which has the nearest knight, 2, in direction `>`, and answers `>` because it is a knight. And so on.

Constrains

$2 \leq n \leq 10^9$.

$2 \leq m \leq \min(n, 100)$.

The maximum amount of queries that can be made is $100 \cdot (m + 1)$ (giving the answer does not count as a query).

Subtasks

1. (11 points) $n \leq 300$.

2. (20 points) $m = 2, n \leq 10^4$, there are no two knights in consecutive positions, and people in the first and last postions of the row are knaves.

3. (20 points) $m = 2$.

4. (21 points) $n \leq 10^4$.

5. (22 points) There are no two knights in consecutive positions, and people in the first and last postions of the row are knaves.

6. (6 points) No additional constraints.