

Problem D. Hidden k -Tuples

This is an interactive problem.

There is a hidden partition of the integers $1, 2, \dots, n$ into $\frac{n}{k}$ disjoint k -tuples. It is guaranteed that $k \mid n$. You need to determine all hidden k -tuples through queries.

In each query, you may choose a set $S \subseteq \{1, 2, \dots, n\}$, and the interactor will return an integer indicating how many hidden k -tuples are **entirely contained** in S .

The number of queries must not exceed $n \lceil \log_2 n \rceil$.

Input

At the start of your program, the interactor outputs a line with two integers n and k .

Here $2 \leq n \leq 300$, $2 \leq k \leq n$, and $k \mid n$.

The hidden partition is stored by the interactor and is not given directly.

After each valid query you output, the interactor returns a line with a single integer r — the number of hidden k -tuples entirely contained in the set you queried.

Output

You may make queries of the following form:

$$? \ c \ x_1 \ x_2 \ \dots \ x_c$$

where $0 \leq c \leq n$, and x_1, x_2, \dots, x_c must be pairwise distinct integers satisfying $1 \leq x_i \leq n$.

This query means you choose the set $S = \{x_1, x_2, \dots, x_c\}$. The interactor returns an integer r — the number of hidden k -tuples entirely contained in S .

When you have determined the answer, output:

$$! \ a_1 \ a_2 \ \dots \ a_n$$

where a_i denotes the tuple number that element i belongs to. The numbering must satisfy $1 \leq a_i \leq \frac{n}{k}$. Two elements in the same hidden tuple must have the same number; two elements in different hidden tuples must have different numbers. The ordering of tuple numbers is arbitrary.

The number of queries must not exceed $n \lceil \log_2 n \rceil$. After outputting the final answer, your program should terminate immediately.

Note that after each query or the final answer, you must flush the output buffer. For example, in C++ you can use:

```
fflush(stdout);
```

or

```
cout << flush;
```

If your output format is invalid, the number of queries exceeds the limit, or the final answer is wrong, you will receive Wrong Answer or Presentation Error.

The interactor is non-adaptive, meaning all k -tuples are fixed before the interaction begins and do not change based on your queries.

Example

| standard input | standard output |
|----------------|-----------------|
| 6 2 | ? 3 1 3 5 |
| 1 | ? 4 2 3 4 5 |
| 1 | ? 6 1 2 3 4 5 6 |
| 3 | ! 1 2 1 3 3 2 |

Note

In the example below, $n = 6$, $k = 2$, and the hidden tuples are $\{1, 3\}$, $\{2, 6\}$, $\{4, 5\}$.

| Program | Interactor |
|-----------------|------------|
| | 6 2 |
| ? 3 1 3 5 | 1 |
| ? 4 2 3 4 5 | 0 |
| ? 6 1 2 3 4 5 6 | 3 |
| ! 1 2 1 3 3 2 | |

Query $\{1, 3, 5\}$: tuple $\{1, 3\} \subseteq S$, so the answer is 1.

Query $\{2, 3, 4, 5\}$: tuple $\{4, 5\} \subseteq S$, so the answer is 1.

Query $\{1, 2, 3, 4, 5, 6\}$: all three tuples are contained, so the answer is 3.

Output ! 1 2 1 3 3 2 means: elements 1, 3 form group 1; elements 2, 6 form group 2; elements 4, 5 form group 3.