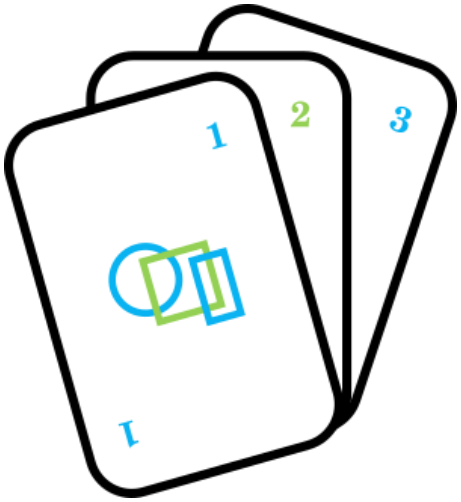
Problem A - As Cartas da Alice

Check our instructions page for detailed information on the qualification and the format of this problem.

Alice, well-known around the world for her card tricks, needs an assistant for her next performance. When starting her trick, Alice shows the audience a deck with N cards, numbered from 1 to N and set in increasing order. Alice is in charge of handling the cards and shuffling them around, but you are left with an equally important task: determining the order of the cards by the end of the trick.



Part I

The simplest operation Alice uses in her tricks is called a *swap*. A swap consists of picking two cards from the deck in different positions A and B, and swapping them. For instance, if a deck with 6 cards is currently ordered as 1 6 4 5 2 3 and Alice performs a swap between positions $A = 3$ and $B = 5$, then the resulting deck will be ordered as 1 6 2 5 4 3.

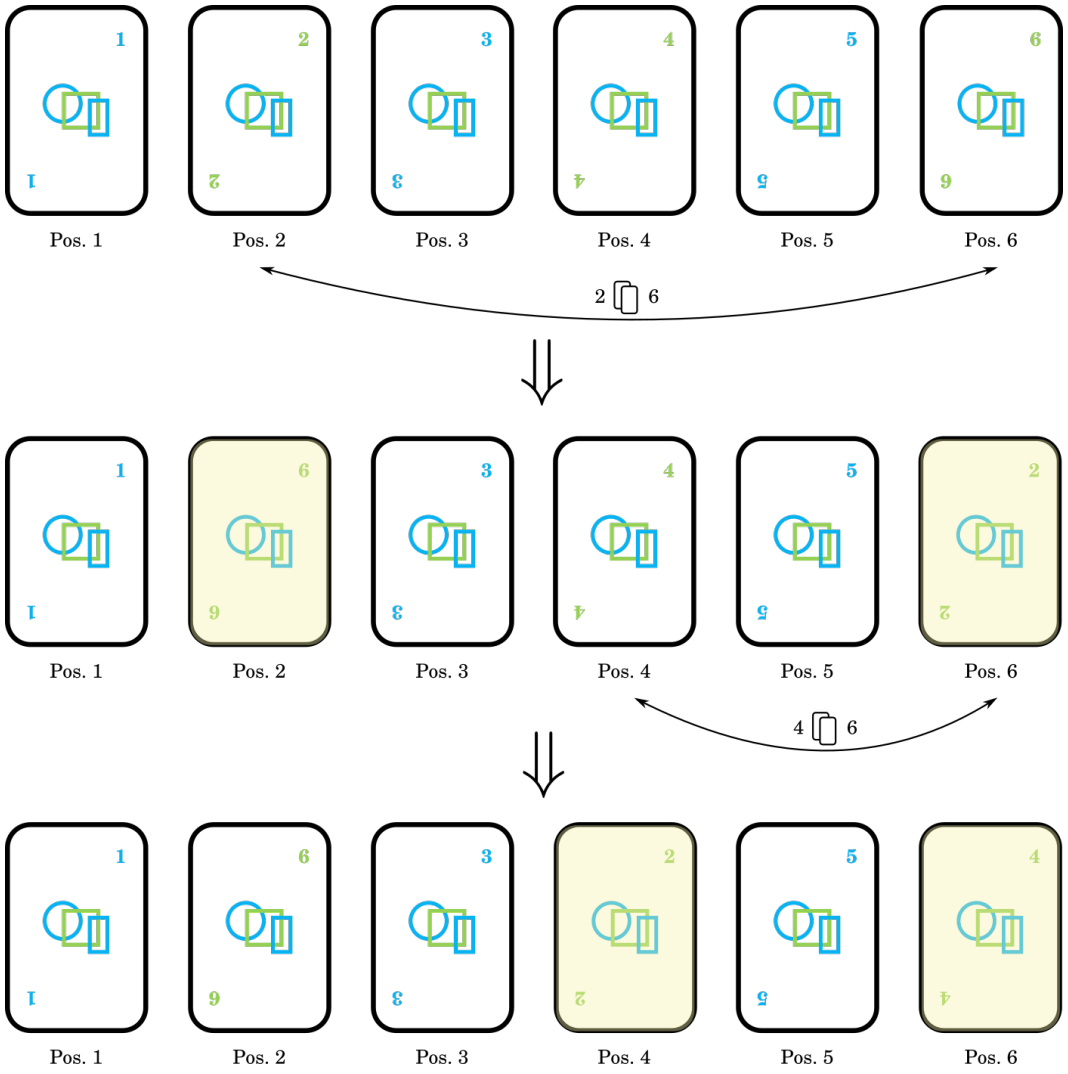
Thus, given the size of the deck N, the number of swaps K, and the positions of the swaps (given by the numbers $A_1, B_1, A_2, B_2, \ldots$), you will have to determine the deck's final order. (See the Input Format section for more information on how these values will be provided.)

Example

For instance, if Alice uses a deck with $N = 6$ cards and performs $K = 2$ swaps, the first between positions $A_1 = 2$ and $B_1 = 6$, and the second between positions $A_2 = 4$ and $B_2 = 6$, we will have the following process:

- First, Alice takes the second and the sixth cards, and swaps their positions.
- Then, with the deck order obtained in the previous step, she swaps the fourth and the sixth cards.
- Finally, the resulting deck is ordered as 1 6 3 2 5 4.

The following picture illustrates this process.



Constraints

The following limits are guaranteed for all test cases of this Part that will be given to the program:

$1 \leq \boldsymbol{N} \leq 10^3$ Deck size
$1 \leq \boldsymbol{K} \leq 10^3$ Number of swap operations

The test cases of this Part of the problem are organized into two groups:

Subtask	Points	Additional Constraints
1	20	$\boldsymbol{N} \leq 10$
2	30	No further constraints

Part II

The other operation that Alice can use in her tricks is the *shuffle*. A shuffle consists of taking the first $\boldsymbol{X}$ cards from the deck and then interleaving them with the remaining ones (see the example

below for more information). For instance, if the deck has $N = 7$ cards, with the deck order 1 2 3 4 5 6 7, and a shuffle with $X = 3$ is performed, then we obtain the order 1 4 2 5 3 6 7. The value of X is always at most $N/2$

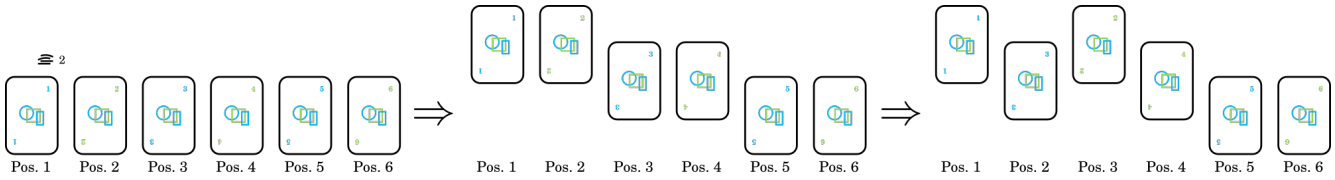
Thus, given the deck size N, the number of shuffles K, and how many cards are shuffled in each one (given by the numbers $X_1, X_2, \ldots$), you have to determine the final deck order. (See the Input Format section for more information on how these values will be provided.)

Example

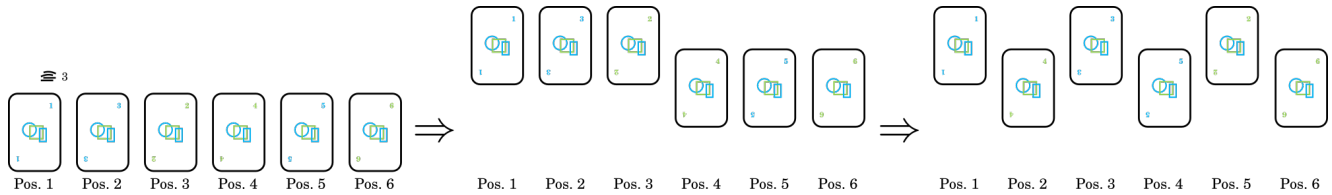
Consider the following example, where the deck has $N = 6$ cards and $K = 2$ shuffle operations are performed, with $X_1 = 2$ and $X_2 = 3$.

- First, Alice takes the first two cards of the deck in the order 1 2 3 4 5 6, and interleaves them with the remaining ones.
- Next, she takes the first three cards of the deck in the newly obtained order, 1 3 2 4 5 6, and repeats the process.
- In the end, the deck is ordered as 1 4 3 5 2 6.

The following image illustrates the result of applying the first *shuffle*:



The following image illustrates the result of applying the second *shuffle*:



Constraints

The following limits are guaranteed for all test cases of this Part that will be given to the program:

$1 \leq N \leq 10^3$ Deck size
$1 \leq K \leq 10^3$ Number of shuffle operations

The test cases of this Part of the problem are organized into two groups:

Subtask	Points	Additional Constraints
3	20	$N \leq 10$
4	30	No further constraints

Summary of Subtasks

The test cases for the problem are organized into four groups with different additional restrictions:

Subtask	Points	Part	Additional Constraints
1	20	Part I	$N \leq 10$
2	30	Part I	No further constraints
3	20	Part II	$N \leq 10$
4	30	Part II	No further constraints

Input Format

The first line contains an integer P which represents the Part of this test case. If it's equal to 1, then the test case refers to Part I, while, if it's equal to 2, then it refers to Part II.

Then follows a line with two integers separated by spaces, the first being N, the number of cards in the deck, and the next being K, the number of operations being performed.

The next part of the input depends on the test case's Part.

In a test case from Part I, this is followed by K lines, each one with two integers separated by spaces, A_i and B_i, the positions of the cards being swapped in the i-th swap operation.

In a test case from Part II, this is followed by K lines, each one with an integer, X_i, the number of cards being shuffled in the i-th shuffle operation. It is guaranteed that the value of the X_i is always at most $N/2$.

Output Format

The output must contain N lines, each one with one integer representing the number of the card in that respective position, in the deck's final order.

Example 1 Input

```
1
6 2
2 6
4 6
```

Example 1 Output

```
1
6
3
2
5
4
```

Example 1 Description

This example corresponds to the example mentioned in Part I of the problem statement.

Example 2 Input

```
2
6 2
2
3
```

Example 2 Output

```
1
4
3
5
2
6
```

Example 2 Description

This example corresponds to the example mentioned in Part II of the problem statement.

Organizers