B. Permutation Game

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

n children are standing in a circle and playing a game. Children's numbers in clockwise order form a permutation $a_1, a_2, ..., a_n$ of length n. It is an integer sequence such that each integer from 1 to n appears exactly once in it.

The game consists of m steps. On each step the current leader with index i counts out a_i people in clockwise order, starting from the next person. The last one to be pointed at by the leader becomes the new leader.

You are given numbers $l_1, l_2, ..., l_m$ — indices of leaders in the beginning of each step. Child with number l_1 is the first leader in the game.

Write a program which will restore a possible permutation $a_1, a_2, ..., a_n$. If there are multiple solutions then print any of them. If there is no solution then print `-1`.

Input

The first line contains two integer numbers n, m $(1 \le n, m \le 100)$.

The second line contains m integer numbers $l_1, l_2, ..., l_m$ $(1 \le l_i \le n)$ — indices of leaders in the beginning of each step.

Output

Print such permutation of n numbers $a_1, a_2, ..., a_n$ that leaders in the game will be exactly $l_1, l_2, ..., l_m$ if all the rules are followed. If there are multiple solutions print any of them.

If there is no permutation which satisfies all described conditions print `-1`.

Examples

input
4 5 2 3 1 4 4

output
3 1 2 4

input
3 3 3 1 2

output
-1

Note

Let's follow leadership in the first example:

- Child 2 starts.
- Leadership goes from 2 to $2 + a_2 = 3$.
- Leadership goes from 3 to $3 + a_3 = 5$. As it's greater than 4, it's going in a circle to 1.
- Leadership goes from 1 to $1 + a_1 = 4$.
- Leadership goes from 4 to $4 + a_4 = 8$. Thus in circle it still remains at 4.