Card game
1 second, 512 MB

In a strange card game, there is a list of **N** cards (1<=**N**<=100,000), each card contains a positive integer, referred to as the *value* of that card. There is another sequence of **M** (1<=**M**<**N**) positive integers: $\mathbf{x_1}$, $\mathbf{x_2}$, $\mathbf{x_3}$, …, $\mathbf{x_M}$, each of value **at most 10**. The game also keeps a position of the "current card," initially the first card in the list.

The game proceeds in **M** rounds. Each round **i**, for 1<=**i**<=**M**, starting from the *current* card, keep counting the cards until the total values is at least $\mathbf{x_i}$. The last card counted will then be discarded from the list. The card after the discarded card would be the current card in the next round. Moreover, if you count to the last card, you continue your counting on the first card in the list.

It is guaranteed that at the end, after **M** rounds, there will be some card left on the list. Also, by following the steps described here, in each round you will never count a single card more than once.

Consider the following example, where **N** = 10, and **M** = 5. Initially, suppose that we have the cards with the following values:

1, 2, 5, 3, 4, 9, 10, 5, 1, 7

and the sequence with 5 positive integers is:

10, 4, 10, 9, 2

Initially, the current card is the first one (shown in a bracket):

[1], 2, 5, 3, 4, 9, 10, 5, 1, 7

The game would proceed like this:

In the first round, $\mathbf{x_1}$=10 starting from the current card, you count the cards: 1, 2, 5, and 3 (with the total of 11>10). Card 3 will be discard and the current card will be card with value 4 on it. The list becomes:

1, 2, 5, **[4]**, 9, 10, 5, 1, 7

In the second round, $\mathbf{x_2}$=4 starting from the current card, you can count only one card: 4 (with the total of 4=4). Card 4 will be discard and the current card will be card with value 9 on it. The list becomes:

1, 2, 5, **[9]**, 10, 5, 1, 7

In the third round, $\mathbf{x_3}$=10 starting from the current card, you can count two cards: 9 and 10 (with the total of 19>=10). Card 10 will be discard and the current card will be card with value 5 on it. The list becomes:

1, 2, 5, 9, **[5]**, 1, 7

In the forth round, $\mathbf{x_4}$=9 starting from the current card, you can count three cards: 5, 1, and 7 (with the total of 13>=9). Card 7 will be discard and the current card will be card with value 1 on it (back to the first card). The list becomes:

[1], 2, 5, 9, 5, 1

In the fifth round, $\mathbf{x_5}$=2 starting from the current card, you can count two cards: 1, and 2 (with the total of 3>=2). Card 2 will be discard and the current card will be card with value 5 on it. The final list becomes:

1, 5, 9, 5, 1

And this is what your program should output.

Your task
You have to write a program that reads the list of values of the cards and the sequence of values $\mathbf{x_i}$, and outputs the final list of remaining card values.

Input
The first line contains two integers N and M (1<=**N**<=100,000; 1<=**M**<**N**). The next line contains **N** positive integers, representing the values of the cards. The values are between 1 and 1,000. The next **M** lines give the sequence $\mathbf{x_i}$. More specifically, line 2+**i** contains $\mathbf{x_i}$. (1<=$\mathbf{x_i}$<=10)

Output
Your program should output **N** – **M** lines, as a list containing the values of the remaining cards.

Subtasks
- Subtask 1 (30%): N <= 1,000
- Subtask 2 (70%): no additional constraints

Example

Input	Output
10 5 1 2 5 3 4 9 10 5 1 7 10 4 10 9 2	1 5 9 5 1

Hints: If you use singly linked lists, to make sure you can delete the current card, it might be useful to keep the pointer to the element in the list ***before*** the current one.