Alice and Bob are playing the following game:

They are given a sequence of ***N*** positive integers with values *less than or equal to* ***N***. The elements of the sequence are numbered from 1 to ***N.*** Equal numbers may exist in the sequence. A set ***S*** is created in the beginning of the game, containing the first ***P*** elements of the sequence. Note that ***S*** may be a *multiset* – it may contain equal elements. The players take turns to play and Alice is playing first. Each move is made as follows:

1) The player whose turn has come, selects one number from the set ***S*** and takes it away, adding its value to his/her score (initially, the score of both players is 0).
2) The next number in the sequence, if any left at all, is added to the set ***S*** (if the sequence is already empty, this action is skipped). This is to say, that after the first taking from ***S***, the number indexed with ***P*+1** is added to the set, after the second one – the number indexed with ***P*+2** is added, etc.

The game continues, until the set ***S*** becomes empty. We assume that each player does their best to maximize their own score. *The **game's result** is the number obtained by subtracting the points, collected by Bob, from those, collected by Alice*.

Task

Write a program **game**, which has to process ***K*** games on a given starting sequence.

Input

Two space separated positive integers ***N*** and ***K*** are read from the first line of the standard input.

The second line consists of ***N*** space separated positive integers a_1, a_2, …., a_N, representing the elements of the given sequence.

The third line contains ***K*** space separated positive integers p_1, p_2, ..., p_K, each defining the starting set ***S***, created from the given sequence (taking the first p_i elements) and intended for the *i*-th game, *i* = 1, 2, ..., ***K***.

Output

The program should print to the standard output ***K*** lines, each containing a single integer – the corresponding **game's result**. Line number ***i*** should contain the result of the game number ***i*** (the games are numbered from 1 to ***K*** by the input).

Constraints

- $1 \le N \le 100\,000$
- $1 \le K \le 2\,000$
- $K \le N$
- $1 \le a_i \le N$ for $i = 1, 2, \ldots, N$
- $1 \le p_i \le N$ for $i = 1, 2, \ldots, K$

- **In 10% of the tests:** $1 \le N \le 10$
- **In 30% of the tests:** $1 \le N \le 600$
- **In 50% of the tests:** $1 \le N \le 10\,000$, $1 \le K \le 1\,000$

Example

Sample input	*Sample output*
5 2 2 4 2 3 5 4 3	2 6

Explanation: *The input data determines that your program will process two games. For both games, the given sequence is the same, but for the first game P = 4 and the starting multiset S is {2, 4, 2, 3}, and for the second game, P=3 and S is {2, 4, 2}.*