An unknown array *x* consists of N integers. The *K-summary* of that array is obtained by dividing the array into segments of length K and summing up the elements in each segment. If N is not divisible by K, the last segment of the division will have less than K elements.

In other words, the K-summary is an array where the elements are, respectively: (*x*[1] + … + *x*[K]), (*x*[K+1] + … + *x*[2K]), and so on, where the last sum that contains *x*[N] can have less than K summands. For example, the 5-summary of an array of 13 elements has three elements (sum of elements 1.-5., sum of elements 6.-10., sum of elements 11.-13.).

It is clear that we cannot reconstruct the elements of the original array from the K-summary, but that might be possible if we knew several K-summaries for different Ks. Write a program that will, given length N and set K_1, K_2, …, K_M, predict how many elements of the original array we would be able to uniquely determine if we knew all the K_i-summaries of the array. (It is not difficult to show that the number of reconstructed elements is independent of the content of the summaries.)

INPUT

The first line contains the integers *N* and *M* ($3 <= N <= 10^9$, $1 <= M <= 10$), the array length and the number of K-summaries.
The second line contains distinct integers K_1, K_2, …, K_M ($2 <= K_i < N$) from the task.

OUTPUT

You must output the required number of reconstructed elements.

SCORING

In test cases worth 40% of total points, it will hold N <= 5 000 000.

SAMPLE TESTS

input	input	input
3 1 2	6 2 2 3	123456789 3 5 6 9
output	**output**	**output**
1	2	10973937

Clarification of the first example: We can determine one element: x[3].
Clarification of the second example: We can determine x[3] and x[4].