

Permutations and Queries

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 512 megabytes

You are given a permutation p of length n . A permutation of length n is an array consisting of n distinct integers from 1 to n in arbitrary order. We define the *cost* of the permutation as the sum over all i from 1 to n of the quantity $(p_i)^i$ (the i -th element of the permutation raised to the power of i). Thus, the *cost* of the permutation p is given by

$$\sum_{i=1}^n (p_i)^i$$

There are q queries of three types:

1. Reverse. After this, your permutation p is replaced by the permutation q , such that $q_i = p_{n-i+1}$ for all i from 1 to n .
2. Invert. After this, your permutation p is replaced by the permutation q , such that $q_i = n - p_i + 1$ for all i from 1 to n .
3. Take the inverse. After this, your permutation p is replaced by the permutation q , such that $q_{p_i} = i$ for all i from 1 to n .

Note that after each operation, p remains a permutation.

After each query, you need to output the *cost* of the permutation.

Input

The first line contains two integers n and q ($1 \leq n, q \leq 100\,000$) — the length of the permutation and the number of queries.

The second line contains n positive integers p_1, p_2, \dots, p_n ($1 \leq p_i \leq n$) — the elements of the permutation. It is guaranteed that all p_i are distinct.

The third line contains q positive integers b_1, b_2, \dots, b_q ($1 \leq b_i \leq 3$) — the description of the queries. The number b_i means that the i -th modification query to be applied to the permutation is of type b_i .

Output

Output q numbers, the i -th of which is the remainder of the *cost* of the permutation modulo 998 244 353, after applying the first i queries.

Examples

standard input	standard output
5 5 1 2 3 4 5 1 2 3 1 2	65 3413 3413 65 3413
5 6 5 3 1 4 2 3 3 1 2 3 1	293 303 3225 215 317 3209

Note

Let's analyze the second example.

Initially, $p = [5, 3, 1, 4, 2]$.

The first query is of type 3, meaning take the inverse. The permutation after this query becomes $[3, 5, 2, 4, 1]$. The *cost* of this permutation is $3^1 + 5^2 + 2^3 + 4^4 + 1^5 = 3 + 25 + 8 + 256 + 1 = 293$.

The second query is of type 3, meaning take the inverse. The permutation after this query becomes $[5, 3, 1, 4, 2]$. The *cost* of this permutation is $5^1 + 3^2 + 1^3 + 4^4 + 2^5 = 5 + 9 + 1 + 256 + 32 = 303$.

The third query is of type 1, meaning reverse. The permutation after this query becomes $[2, 4, 1, 3, 5]$. The *cost* of this permutation is $2^1 + 4^2 + 1^3 + 3^4 + 5^5 = 3225$.

The fourth query is of type 2, meaning invert. The permutation after this query becomes $[4, 2, 5, 3, 1]$. The *cost* of this permutation is $4^1 + 2^2 + 5^3 + 3^4 + 1^5 = 215$.

The fifth query is of type 3, meaning take the inverse. The permutation after this query becomes $[5, 2, 4, 1, 3]$. The *cost* of this permutation is $5^1 + 2^2 + 4^3 + 1^4 + 3^5 = 317$.

The last query is of type 1, meaning reverse. The permutation after this query becomes $[3, 1, 4, 2, 5]$. The *cost* of this permutation is $3^1 + 1^2 + 4^3 + 2^4 + 5^5 = 3209$.

Scoring

The tests for this problem consist of five groups. Points for each group are awarded only if all tests in the group and all tests in some of the previous groups are passed. Note that passing the tests from the statement is not required for some groups. The final score for each group is the maximum score obtained for that group of tests across all submitted solutions.

Group	Points	Additional constraints		Required groups	Comment
		n	q		
0	0	–	–	–	Tests from the statement.
1	15	$n \leq 1000$	$q \leq 1000$	0	
2	22	–	–	–	$b_i = b_j$ for all $1 \leq i, j \leq q$
3	26	–	–	–	$b_i \leq 2$ for all $1 \leq i \leq q$
4	16	–	–	–	$p_i = i$ for all $1 \leq i \leq n$
5	21	–	–	0 – 4	