Total Ambiguity

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

We call an array `ambiguous` if there exists **strictly more** than one way to split it into two non-empty subarrays such that both of these subarrays are permutations. Note that each element of the array must belong to exactly one subarray of the split, and each subarray must consist of consecutive elements of the original array.

For example, the array $[1, 2, 1]$ is `ambiguous`; it can be split into $[1] + [2, 1]$ as well as $[1, 2] + [1]$.

You are given an array $a_1, a_2, \ldots, a_n$. In one operation, you can change any element of the array to any integer from 1 to n. What is the minimum number of operations required to make the array $a_1, a_2, \ldots, a_n$ ambiguous?

Recall that an array $p_1, p_2, \ldots, p_n$ is called a permutation if for each x from 1 to n, there exists exactly one i such that $p_i = x$.

Input

The first line of input contains an integer n ($3 \le n \le 10^7$) — the length of the array a and an integer f, which describes the format in which the array a will be given.

If $f = 1$, the format is standard, and then:

The second line of input contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le n$) — the elements of the array a.

If $f = 2$, the format is as follows:

The second line of input contains an integer m ($3 \le m \le \min(n, 2 \cdot 10^5)$) and three integers x, y, z ($0 \le x, y, z < n$).

The third line of input contains m integers $b_1, b_2, \ldots, b_m$ ($1 \le b_i \le n$).

The values $a_1, a_2, \ldots, a_n$ — the array for which you need to find the answer to the problem, are calculated as follows:

- For $1 \le i \le m$: $a_i = b_i$
- For $m < i \le n$: $a_i = 1 + (x \cdot a_{i-2} + y \cdot a_{i-1} + z) \bmod n$

Output

In a single line, output one number — the answer to the problem.

Scoring

Subtask	Points	Additional Constraints				Required Subgroups
		n	a	f	Additional	
1	15	$n \leq 7$	–	$f = 1$	–	
2	7	$n \leq 2000$	–	$f = 1$	The answer does not exceed 1	
3	21	$n \leq 2000$	–	$f = 1$	–	1,2
4	4	$n \leq 2 \cdot 10^5$	$a_i = i$	$f = 1$	–	
5	4	$n \leq 2 \cdot 10^5$	$a_i = 1$	$f = 1$	–	
6	15	$n \leq 2 \cdot 10^5$	–	$f = 1$	The answer does not exceed 1	2
7	18	$n \leq 2 \cdot 10^5$	–	$f = 1$	–	1-6
8	16	–	–	$f = 2$	–	

Examples

standard input	standard output
7 1 2 2 1 4 3 7 7	3
4 1 1 2 2 1	1
3 1 1 1 1	1
3 1 1 2 3	1
15 2 6 12 5 11 10 9 8 7 1 1	6

Note

In the first test case, the following three operations can be performed:

- Replace the first element of the array with 3
- Replace the sixth element of the array with 2
- Replace the seventh element of the array with 1

After the operations, the array looks like this: $[3, 2, 1, 4, 3, 2, 1]$, and it is easy to see that this array is ambiguous.