Codeforces Round #239 (Div. 1)

A. Triangle

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

There is a right triangle with legs of length a and b. Your task is to determine whether it is possible to locate the triangle on the plane in such a way that none of its sides is parallel to the coordinate axes. All the vertices must have integer coordinates. If there exists such a location, you have to output the appropriate coordinates of vertices.

Input

The first line contains two integers a, b ($1 \le a, b \le 1000$), separated by a single space.

Output

In the first line print either "YES" or "NO" (without the quotes) depending on whether the required location exists. If it does, print in the next three lines three pairs of integers — the coordinates of the triangle vertices, one pair per line. The coordinates must be integers, not exceeding 10^9 in their absolute value.

Examples

input
1 1
output
NO

input
5 5
output
YES 2 1 5 5 -2 4

input
5 10
output
YES -10 4 -2 -2 1 2

B. Long Path

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

One day, little Vasya found himself in a maze consisting of $(n + 1)$ rooms, numbered from 1 to $(n + 1)$. Initially, Vasya is at the first room and to get out of the maze, he needs to get to the $(n + 1)$-th one.

The maze is organized as follows. Each room of the maze has two one-way portals. Let's consider room number i $(1 \le i \le n)$, someone can use the first portal to move from it to room number $(i + 1)$, also someone can use the second portal to move from it to room number p_i, where $1 \le p_i \le i$.

In order not to get lost, Vasya decided to act as follows.

- Each time Vasya enters some room, he paints a cross on its ceiling. Initially, Vasya paints a cross at the ceiling of room 1.
- Let's assume that Vasya is in room i and has already painted a cross on its ceiling. Then, if the ceiling now contains an odd number of crosses, Vasya uses the second portal (it leads to room p_i), otherwise Vasya uses the first portal.

Help Vasya determine the number of times he needs to use portals to get to room $(n + 1)$ in the end.

Input

The first line contains integer n $(1 \le n \le 10^3)$ — the number of rooms. The second line contains n integers p_i $(1 \le p_i \le i)$. Each p_i denotes the number of the room, that someone can reach, if he will use the second portal in the i-th room.

Output

Print a single number — the number of portal moves the boy needs to go out of the maze. As the number can be rather large, print it modulo 1000000007 $(10^9 + 7)$.

Examples

input
2 1 2
output
4

input
4 1 1 2 3
output
20

input
5 1 1 1 1 1
output
62

C. Curious Array

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You've got an array consisting of n integers: $a[1], a[2], ..., a[n]$. Moreover, there are m queries, each query can be described by three integers l_i, r_i, k_i. Query l_i, r_i, k_i means that we should add $\binom{j-l_i+k_i}{k_i}$ to each element $a[j]$, where $l_i \le j \le r_i$.

Record $\binom{y}{x}$ means the binomial coefficient, or the number of combinations from y elements into groups of x elements.

You need to fulfil consecutively all queries and then print the final array.

Input

The first line contains integers n, m ($1 \le n, m \le 10^5$).

The second line contains n integers $a[1], a[2], ..., a[n]$ ($0 \le a_i \le 10^9$) — the initial array.

Next m lines contain queries in the format l_i, r_i, k_i — to all elements of the segment $l_i ... r_i$ add number $\binom{j-l_i+k_i}{k_i}$ ($1 \le l_i \le r_i \le n$; $0 \le k \le 100$).

Output

Print n integers: the i-th number is the value of element $a[i]$ after all the queries. As the values can be rather large, print them modulo 1000000007 ($10^9 + 7$).

Examples

input
5 1 0 0 0 0 0 1 5 0
output
1 1 1 1 1

input
10 2 1 2 3 4 5 0 0 0 0 0 1 6 1 6 10 2
output
2 4 6 8 10 7 3 6 10 15

D. Largest Submatrix 3

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given matrix a of size $n \times m$, its elements are integers. We will assume that the rows of the matrix are numbered from top to bottom from 1 to n, the columns are numbered from left to right from 1 to m. We will denote the element on the intersecting of the i-th row and the j-th column as a_{ij}.

We'll call submatrix i_1, j_1, i_2, j_2 $(1 \le i_1 \le i_2 \le n;\ 1 \le j_1 \le j_2 \le m)$ such elements a_{ij} of the given matrix that $i_1 \le i \le i_2$ AND $j_1 \le j \le j_2$. We'll call the area of the submatrix number $(i_2 - i_1 + 1) \cdot (j_2 - j_1 + 1)$. We'll call a submatrix inhomogeneous, if all its elements are distinct.

Find the largest (in area) inhomogenous submatrix of the given matrix.

Input

The first line contains two integers n, m $(1 \le n, m \le 400)$ — the number of rows and columns of the matrix, correspondingly.

Each of the next n lines contains m integers a_{ij} $(1 \le a_{ij} \le 160000)$ — the elements of the matrix.

Output

Print a single integer — the area of the optimal inhomogenous submatrix.

Examples

input
3 3 1 3 1 4 5 6 2 6 1
output
6

input
3 4 5 2 3 1 3 3 5 3 4 4 4 5
output
4

input
2 6 1 2 3 4 5 6 8 6 7 8 9 1
output
8

E. k-d-sequence

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

We'll call a sequence of integers a good k-d sequence if we can add to it at most k numbers in such a way that after the sorting the sequence will be an arithmetic progression with difference d.

You got hold of some sequence a, consisting of n integers. Your task is to find its longest contiguous subsegment, such that it is a good k-d sequence.

Input

The first line contains three space-separated integers n, k, d ($1 \le n \le 2 \cdot 10^5$; $0 \le k \le 2 \cdot 10^5$; $0 \le d \le 10^9$). The second line contains n space-separated integers: $a_1, a_2, \ldots, a_n$ ($-10^9 \le a_i \le 10^9$) — the actual sequence.

Output

Print two space-separated integers l, r ($1 \le l \le r \le n$) show that sequence $a_l, a_{l+1}, \ldots, a_r$ is the longest subsegment that is a good k-d sequence.

If there are multiple optimal answers, print the one with the minimum value of l.

Examples

input
6 1 2 4 3 2 8 6 2
output
3 5

Note

In the first test sample the answer is the subsegment consisting of numbers 2, 8, 6 — after adding number 4 and sorting it becomes sequence 2, 4, 6, 8 — the arithmetic progression with difference 2.