

Codeforces Round #239 (Div. 2)

A. Line to Cashier

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

Little Vasya went to the supermarket to get some groceries. He walked about the supermarket for a long time and got a basket full of products. Now he needs to choose the cashier to pay for the products.

There are n cashiers at the exit from the supermarket. At the moment the queue for the i -th cashier already has k_i people. The j -th person standing in the queue to the i -th cashier has $m_{i,j}$ items in the basket. Vasya knows that:

- the cashier needs 5 seconds to scan one item;
- after the cashier scans each item of some customer, he needs 15 seconds to take the customer's money and give him the change.

Of course, Vasya wants to select a queue so that he can leave the supermarket as soon as possible. Help him write a program that displays the minimum number of seconds after which Vasya can get to one of the cashiers.

Input

The first line contains integer n ($1 \leq n \leq 100$) — the number of cashes in the shop. The second line contains n space-separated integers: k_1, k_2, \dots, k_n ($1 \leq k_i \leq 100$), where k_i is the number of people in the queue to the i -th cashier.

The i -th of the next n lines contains k_i space-separated integers: $m_{i,1}, m_{i,2}, \dots, m_{i,k_i}$ ($1 \leq m_{i,j} \leq 100$) — the number of products the j -th person in the queue for the i -th cash has.

Output

Print a single integer — the minimum number of seconds Vasya needs to get to the cashier.

Examples

input
1 1 1 1
output
20

input
4 1 4 3 2 100 1 2 2 3 1 9 1 7 8
output
100

Note

In the second test sample, if Vasya goes to the first queue, he gets to the cashier in $100 \cdot 5 + 15 = 515$ seconds. But if he chooses the second queue, he will need $1 \cdot 5 + 2 \cdot 5 + 2 \cdot 5 + 3 \cdot 5 + 4 \cdot 15 = 100$ seconds. He will need $1 \cdot 5 + 9 \cdot 5 + 1 \cdot 5 + 3 \cdot 15 = 100$ seconds for the third one and $7 \cdot 5 + 8 \cdot 5 + 2 \cdot 15 = 105$ seconds for the fourth one. Thus, Vasya gets to the cashier quicker if he chooses the second or the third queue.

B. Garland

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

Once little Vasya read an article in a magazine on how to make beautiful handmade garland from colored paper. Vasya immediately went to the store and bought n colored sheets of paper, the area of each sheet is 1 square meter.

The garland must consist of exactly m pieces of colored paper of arbitrary area, each piece should be of a certain color. To make the garland, Vasya can arbitrarily cut his existing colored sheets into pieces. Vasya is not obliged to use all the sheets to make the garland.

Vasya wants the garland to be as attractive as possible, so he wants to maximize the total area of m pieces of paper in the garland. Calculate what the maximum total area of the pieces of paper in the garland Vasya can get.

Input

The first line contains a non-empty sequence of n ($1 \leq n \leq 1000$) small English letters ("a"..."z"). Each letter means that Vasya has a sheet of paper of the corresponding color.

The second line contains a non-empty sequence of m ($1 \leq m \leq 1000$) small English letters that correspond to the colors of the pieces of paper in the garland that Vasya wants to make.

Output

Print an integer that is the maximum possible total area of the pieces of paper in the garland Vasya wants to get or -1, if it is impossible to make the garland from the sheets he's got. It is guaranteed that the answer is always an integer.

Examples

input
aaabbac aabbccac
output
6

input
a z
output
-1

Note

In the first test sample Vasya can make an garland of area 6: he can use both sheets of color b , three (but not four) sheets of color a and cut a single sheet of color C in three, for example, equal pieces. Vasya can use the resulting pieces to make a garland of area 6.

In the second test sample Vasya cannot make a garland at all — he doesn't have a sheet of color Z .

C. 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 ($1 \leq a, b \leq 1000$), 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

D. 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 ($1 \leq i \leq n$), 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 $1 \leq p_i \leq i$.

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 ($1 \leq n \leq 10^3$) — the number of rooms. The second line contains n integers p_i ($1 \leq p_i \leq i$). 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

E. 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: $a[1], a[2], \dots, a[n]$. 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 $\binom{r-l_i+k_i}{k_i}$ to each element $a[j]$, where $l_i \leq j \leq r_i$.

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 ($1 \leq n, m \leq 10^5$).

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

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

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