Problem A. LIS and GCD

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 megabytes

You are given two arrays a and b. Each array has n integer numbers. You need to compute the sum of GCDs of all subsequences of array b such that the corresponding subsequences of array a are increasing.

Input

The first input line contains one integer number n. The second line contains n integer numbers a_i. The third line contains n integer numbers b_i.

$$1 \le n \le 10^5$$
$$1 \le a_i \le 10^9$$
$$1 \le b_i \le 10^6$$

Output

Display the answer modulo $10^9 + 7$.

Examples

standard input	standard output
2 1 2 1 2	4
2 2 1 1 2	3
4 2 3 1 4 2 3 6 9	29

Note

Third example: Here are examples of increasing subsequences from a and the corresponding GCD values: 2->2, 3->3, 1->6, 4-> 9, 2,3->1, 2,4->1, 3,4->3, 1,4->3, 2,3,4->1.

Problem B. Sum-and-sum

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 megabytes

Little Lyosha has an array, which consists of n integers. Every day he selects a certain continuous segment of the array and carefully examines each of its subsegments. For every subsegment he calculates bitwise AND of its elements, adds the sum of all elements of the subsegment to it and then writes down the calculated value into his Notebook of Important Numbers. Note that numbers in the array can be negative, so when calculating bitwise AND one should use their absolute values. At the end of the day, after Lyosha has carefully examined all the subsegments, he looks through all the written values, takes the maximal number among them and then writes it down into his Notebook of Very Important Numbers. The Notebook of Very Important Numbers is naturally very important for Lyosha, so he wants to make sure that there are no errors. Help the little boy to check his notes.

Input

The first line contains two integers n and m — the length of the array and the number of notes Little Lyosha wants to check. The second line contains n integers a_i — elements of the array. Then m lines follow. The i-th line contains integers l_i and r_i, which means that on the i-th day the boy played with segment $[l_i, r_i]$ of the array.

$$1 \le n \le 3 \times 10^5$$
$$1 \le m \le 3 \times 10^4$$
$$|a_i| \le 10^5$$
$$1 \le l_i \le r_i \le n$$

Output

Print m lines. On the i-th line print the number that Lyosha should have written in his notebook at the i-th day.

Example

standard input	standard output
7 3 3 -3 3 -3 6 5 -3 2 4 3 7 2 6	6 15 15

Problem C. A lost array

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	64 megabytes

Fortune smiled upon little Lyosha: he managed to get a very rare and unique array A, which consists of n integers. Having obtained such a precious object, the boy immediately brought it to his underground laboratory so as to fathom the mystery of the array's structure. As an experienced researcher, Lyosha decided not to hurry with performing all the horrible experiments with the poor array, but to observe it at first. On a piece of paper he made some notes on m observations of the array. Each observation is described by 3 numbers l_i, r_i and s_i, which mean, that the sum of elements modulo 2 on the subsegment $[l_i, r_i]$ of the array equals s_i. Lyosha has prepared everything required for the $(m+1)$-th observation, but at that point he heard his mother calling him for dinner. Once the boy returned to his lab after the dinner, he shuddered with horror: the array was gone. At all. All the hopes of the little boy of getting the Nobel Prize in the field of array research have vanished in no time. However Lyosha remembered, that he still had some information left about the array. Based on the facts he had previously written down, he decided to build a virtual prototype of the array and conduct experiments with it. Of course, the little boy realized there was no way he could restore an identical copy of the lost array, because there are infinitely many arrays which comply with his previously written down observations. Certainly, all of us know, that operations with huge numbers take huge amount of time, and things get even worse when it comes to negative ones. In order to make his computer prototype as fast as possible, little Lyosha decided to build an array that complies with his notes, its elements are all non-negative and their sum is as small as possible. Help him to do it!

Input

The first line of the input contains two integers n and m — the size of the lost array and the number of observations, which little boy made about it. Then m lines follow. The i-th line contains three integers l_i, r_i and s_i - description of the i-th observation.

$$1 \le n \le 40$$

$$0 \le m \le 100$$

$$1 \le l_i \le r_i \le n$$

$$0 \le s_i \le 1$$

$$0 \le a_i \le 10^9$$

Output

Print n integers a_i - elements of the brand new array. If there are several possible arrays, print lexicographically minimal of them. Formally, an array X is lexicographically less then array Y if there is such number $k \le |X|$, that for all $i < k$ $X_i = Y_i$ and $X_k < Y_k$.

Example

standard input	standard output
3 3 2 2 1 3 3 0 2 3 1	0 1 0

Problem D. Bulls and cows

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	64 megabytes

You woke up on the chilly tiled floor of a farm. "Hello... said a spooky voice. "I want to play a game."You did not have a choice, so you listened to the game's rules.

The voice knows a 4-digit secret number, all digits of which are different. You need to guess it and for that you can make several tries. During each try you say a 4-digit number, all digits of which are different. The voice tells you the number of matches. If the matching digits are in their right positions, they are called "bulls"; if in different positions, they are "cows".

For example: the secret number is 3219. Your try is 2310 and the voice's answer is 12: one bull (the digit 1 is on the right position) and two cows (digits 2 and 3 are present in the number, but they are in different positions).

You have at most 8 tries to guess the secret number. Once you guess it, you can leave the crazy farmer's place alive. Good luck!

Input

For each your try you will receive a line of two digits: number of bulls and number of cows.

Output

Don't forget to put a newline character and to flush the output stream after you print your guess. For example fflush(stdout) in C++, System.out.flush() in Java, and flush(output) in Pascal.

Example

standard input	standard output
0123	11
0145	11
0256	11
0612	30
0178	11
0619	40

Problem E. A+B

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

You are given N numbers A_i. You are required to find the sum of A_i. Unfortunately, the numbers in the input were mixed up in such way that the number N is not necessarily the first number of the input.

Input

The first line contains the numbers X_i ($0 \le X_i \le 50$). The amount of numbers in the input is not greater than 50. Among these numbers are the number N and the numbers A_i.

Output

Calculate the sum of A_i.

Example

standard input	standard output
23 3 12 7	42

Problem F. Stone, grass and fire

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Farmer Lyosha owns a rectangular field. The field has n rows and m columns. Field cells positions are denoted by pairs of numbers (i, j), where i is the row index and j is the column index. Each cell of the field has either grass (denoted by character `'.'`) or stone (denoted by character `'X'`).

Disasters happen, and Lyosha's field is on fire. The fire has started in different cells. There are k fire sources. The i-th fire source has started in grass cell (x_i, y_i) and it will burn for c_i seconds. Each second the fire spreads from an already burning cell to adjacent cells that have a common edge with the burning cell. If in current cell fire will burn for c second, then this fire will burn for $c - 1$ seconds in adjacent cells. The fire did not spread to adjacent cells, if it will burn for zero seconds in current cell.

Assume that all fire sources are independent. All fire sources have distinct coordinates. Stone does not burn.

Lyosha is very eager to know how many grass cells are going to burn down.

Input

The first input line contains three integer numbers — n, m, k — which denote the field size and the number of fire sources. Following that are n lines describing the field, each line has m characters. The j-th character in the i-th line denotes type of the (i, j) cell. The next k lines each contain three integer numbers x_i, y_i, c_i that describe the i-th fire source.

$$1 \le N, M \le 500$$

$$0 \le K \le N \times M$$

$$1 \le x_i \le N$$

$$1 \le y_i \le M$$

$$1 \le c_i \le 10^9$$

Output

Output one integer number: the number of grass cells that will burn down.

Example

standard input	standard output
4 5 2 .X... .X.X. .X.X. ...X. 1 3 1 3 3 4	9

Problem G. ReHanoi Towers

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	64 megabytes

Everybody knows the classical puzzle about Hanoi Towers. There are three rods and one of them has n rings in ascending order of radius, the biggest disk is at the bottom, smallest is at the top. The objective of the puzzle is to move all of the rings to another rod in a minimal number of steps and preserve the disk order. During one step you are allowed to move only one ring and smaller radius rings can only be put on top of bigger radius rings.

You are required to solve modified Hanoi Towers puzzle. You have four rods instead of three. All other rules are left unchanged.

Input

A single integer number n — the number of rings on the first rod.

$$1 \le n \le 10^9$$

Output

The minimal number of steps modulo $10^9 + 7$.

Examples

standard input	standard output
2	3
8	33

Problem H. Cipher

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	64 megabytes

Lyosha is tired of inventing a bicycle, so he decided to invent an unbreakable cipher. Mature cryptanalyst Vasya knows that in his algorithm Lyosha uses x and y such that $l_i \le x \le r_i$ and $l_j \le y \le r_j$ for $1 \le i < j \le n$. Help Vasya decipher Lyosha's private message. You need to compute number of tuples (x, y, i, j) such that number of ones in binary representation of $x \oplus y$ is odd.

Here $\oplus$ denotes addition modulo 2 aka exclusive disjunction aka exclusive OR aka xor.

Input

The first line contains number n. Following that are n lines, where the i^{th} line contains pair of numbers l_i r_i.

$$1 \le n \le 10^5$$

$$1 \le l_i \le r_i \le 10^{18}$$

Output

The only output line should contain the number of tuples (x, y, i, j) modulo $10^9 + 7$.

Example

standard input	standard output
4 1 4 7 9 6 7 2 8	42

Problem I. Abacaba Ltd.

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	64 megabytes

Yasha is a young businessman who has read lots of marketing books. He decided to give his company a name that it will be the first in the alphabetical list of organizations of the city. But the city council is tired of all these endless "Abacaba Ltd.s. So they gave Yasha a random string as the name for his company, in which he must do exactly one modification. Modification is done by swapping any two characters such that the distance between them is equal to any of the favorite numbers of the head of the city council. The distance is calculated as absolute difference of positions of characters in the string. It is guaranteed that such modification is possible.

Input

The first line contains string s — the initial random name. It consists of small latin letters. The second line contains one integer number n —the amount of the favorite numbers of the head of the city council. The next line has n integer numbers a_i.

$$2 \leq |S| \leq 10^5$$

$$1 \leq n \leq 50$$

$$1 \leq a_i < |S|$$

Output

Output the name of the company that want to choose.

Example

standard input	standard output
acdbe 2 4 2	abdce

Problem J. Covering distance

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	64 megabytes

You are given a set of n geometric points. Assume arbitrary point P and all possible k non-degenerate different triangles T_i formed by point P and two other points within the set. Two triangles are considered different if they differ by at least one vertex. There are m_i points $t_{i,j}$ situated strictly within T_i (points on the borders are excluded). Let's determine cover distance as $D = \frac{\sum_{i=1}^{k}\sum_{j=1}^{m_i} dist(P,t_{i,j})}{\sum_{i=1}^{k} m_i}$, where $dist(X,Y)$ means Euclidean distance between points X and Y. When there are no applicable points $t_{i,j}$ for point P, then cover distance is equal to -1.

You are required to compute cover distance for each point of the set.

Input

The first line contains number of points n. Following that are n lines, where the i^{th} line contains coordinates of the i^{th} point represented by two integer numbers x_i y_i. It is guaranteed that all points are distinct.

$$1 \le n \le 300$$
$$|x_i|, |y_i| \le 10^9$$

Output

For each point p output the required average distance. If no point satisfies the above requirements, output -1. Your output should have an absolute or relative error of at most 10^{-9}.

Example

standard input	standard output
6 0 2 2 4 -1 3 0 0 1 3 1 2	-1 1.885618083 1.648528137 2.290569415 1.414213562 1

Problem K. Problem

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	64 megabytes

Tired of long, confusing, boring, tedious fairy-tales and legends in problem statements? All these pathetic authors' attempts to express their hidden literary talents, which, in fact, they barely have... This inexplicable desire to mangle a crystal clear, orderly and elegant mathematical problem and throw it into our dreary reality... Enough! This problem will be short, clear and unequivocal.

You are given array F, which consists of n integers, and m quadruples a_i, b_i, c_i, d_i, such that $a_i \le b_i$, $c_i \le d_i$. For every quadruple compute and print value ans_i that is equal to the number of ordered pairs (u, v), such that $a_i \le u \le b_i$, $c_i \le v \le d_i$ and $F_u = F_v$.

Input

The first line of the input contains two integers n and m — length of array F and the number of quadruples. The second line contains n space-separated integers F_i. Each of next m lines contains four integers — a_i, b_i, c_i and d_i.

$$1 \le n \le 2 \times 10^5$$

$$1 \le m \le 3 \times 10^4$$

$$0 \le F_i \le 10^5$$

$$1 \le a_i \le b_i \le n$$

$$1 \le c_i \le d_i \le n$$

Output

Print m lines. The i-th line should contain only one integer — ans_i.

Examples

standard input	standard output
10 5 4 1 5 0 1 1 1 0 2 1 7 10 4 6 3 5 3 5 5 10 4 7 9 10 3 10 1 3 2 10	5 3 13 5 6
5 3 2 1 3 3 2 3 3 2 5 3 4 3 4 4 5 2 2	2 4 0