Problem Alpha

Solve this one by emailing sleator@cs.cmu.edu explaining how you would do it, and why your approach will work. He'll email you back the judgement.

Seniorious has n pieces of talisman. Willem puts them in a line, the i-th of which is an integer a_i.

In order to maintain it, Willem needs to perform m operations.

There are four types of operations:

- $1\ l\ r\ x$: For each i such that $l \le i \le r$, assign $a_i + x$ to a_i.
- $2\ l\ r\ x$: For each i such that $l \le i \le r$, assign x to a_i.
- $3\ l\ r\ x$: Print the x-th smallest number in the index range $[l, r]$, i.e. the element at the x-th position if all the elements a_i such that $l \le i \le r$ are taken and sorted into an array of non-decreasing integers. It's guaranteed that $1 \le x \le r - l + 1$.
- $4\ l\ r\ x\ y$: Print the sum of the x-th power of a_i such that $l \le i \le r$, modulo y, i.e. $(\sum_{i=l}^{r} a_i^x) \bmod y$.

Input

The only line contains four integers $n, m, seed, v_{max}$ ($1 \le n, m \le 10^5, 0 \le seed < 10^9 + 7, 1 \le vmax \le 10^9$).

The initial values and operations are generated using following pseudo code:

```
def rnd():

    ret = seed
    seed = (seed * 7 + 13) mod 1000000007
    return ret

for i = 1 to n:

    a[i] = (rnd() mod vmax) + 1

for i = 1 to m:

    op = (rnd() mod 4) + 1
    l = (rnd() mod n) + 1
    r = (rnd() mod n) + 1
    if (l > r):
         swap(l, r)

    if (op == 3):
        x = (rnd() mod (r - l + 1)) + 1
    else:
        x = (rnd() mod vmax) + 1

    if (op == 4):
        y = (rnd() mod vmax) + 1
```

Here op is the type of the operation mentioned in the legend.

A. Arthur's Language

time limit per test: 1.0 s
memory limit per test: 1024 MB
input: standard input
output: standard output

Arthur is an extremely important person and rarely has the time to talk. To minimize the time spent with words, he developed his own language, which he uses on a daily basis. However, it's very hard to understand whatever comes out of his mouth.

In order to improve his interaction with the rest of the world, Arthur has asked the Narratives Industrial Complex (CIN in Portuguese) to develop a program which receives a speech S produced by Arthur and a pattern w and prints how many distinct ways we can obtain w by only removing characters from S. Two ways are considered distinct if there is an index $0 \leq i < |S|$ such as S_i is present in one and not in the other.

Since you are considered the most experienced programmer in CIN, your team has trusted you the task of writing this program.

Input

The first line consists of a string $S(1 \leq |S| \leq 10^5)$, representing Arthur's speech. S contains only lower and upper case letters.

The second and last line consists of a string $w(1 \leq |w| \leq 10)$, the pattern to be searched.

Output

Print a single integer: The number of distinct ways to obtain w by removing letters from S. Because this number can be large, print the remained of this number divided by $1000000007 = 10^9 + 7$

Examples

input
weewrshkim sim
output
1

input
qqqaaabbbcccfffrrr qabcfr
output
729

B Jenny and the Batteries

time limit per test: 2.0 s
memory limit per test: 64 MB
input: standard input
output: standard output

Jenny works at BBB (Battery Bank of Brazil). She's in charge of optimizing the batteries' displacements, but can't solve the problem. Knowing you are a programmer, she asked for your help.

At BBB, the batteries are arranged in piles. In a bank, the batteries can't be discharged. So, when recharging then, the bank spends M energy units, where M is the size of the greatest pile. Each pile initially has a_i batteries.

There is a machine the can move a battery from pile i to pile j with cost $b_i + c_j$. Jenny has a limited budget of K for moving the batteries and the bank wants the minimum M possible at the end.

Input

The first line of input consists of the integers N and K ($1 \leq N \leq 10^5$, $1 \leq K \leq 10^{18}$). The following N lines contain a_i, b_i and c_i ($0 \leq a_i, b_i, c_i \leq 10^6$).

Output

Print the minumum M possible.

Example

input
5 20 5 3 8 2 8 3 3 2 4 7 2 1 6 1 1
output
5

C Ithea Plays With Chtholly

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

This is an interactive problem. Refer to the Interaction section below for better understanding.

Ithea and Chtholly want to play a game in order to determine who can use the kitchen tonight.

Initially, Ithea puts n clear sheets of paper in a line. They are numbered from 1 to n from left to right.

This game will go on for m rounds. In each round, Ithea will give Chtholly an integer between 1 and c, and Chtholly needs to choose one of the sheets to write down this number (if there is already a number before, she will erase the original one and replace it with the new one).

Chtholly wins if, at any time, all the sheets are filled with a number and the n numbers are in non-decreasing order looking from left to right from sheet 1 to sheet n, and if after m rounds she still doesn't win, she loses the game.

Chtholly really wants to win the game as she wants to cook something for Willem. But she doesn't know how to win the game. So Chtholly finds you, and your task is to write a program to receive numbers that Ithea gives Chtholly and help her make the decision on which sheet of paper write this number.

Input

The first line contains 3 integers n, m and c ($n, m \geq 2, 1 \leq c \leq 1000, 1 \leq n \cdot \lceil \frac{c}{2} \rceil \leq m \leq 1000, \lceil x \rceil$ means x rounded up) — the number of sheets, the number of rounds and the largest possible number Ithea can give to Chtholly respectively. The remaining parts of input are given throughout the interaction process.

Interaction

In each round, your program needs to read one line containing a single integer p_i ($1 \leq p_i \leq c$), indicating the number given to Chtholly.

Your program should then output a line containing an integer between 1 and n, indicating the number of sheet to write down this number in.

After outputting each line, don't forget to flush the output. For example:

- `fflush(stdout)` in C/C++;
- `System.out.flush()` in Java;
- `sys.stdout.flush()` in Python;
- `flush(output)` in Pascal;
- See the documentation for other languages.

If Chtholly wins at the end of a round, no more input will become available and your program should terminate normally. It can be shown that under the constraints, it's always possible for Chtholly to win the game.

Example

input
2 4 4 2 1 3
output
1 2 2

Note

In the example, Chtholly initially knew there were 2 sheets, 4 rounds and each number was between 1 and 4. She then received a 2 and decided to write it in the 1st sheet. Then she received a 1 and wrote it in the 2nd sheet. At last, she received a 3 and replaced 1 with 3 in the 2nd sheet. At this time all the sheets were filled with a number and they were non-decreasing, so she won the game.

Note that it is required that your program terminate immediately after Chtholly wins and do not read numbers from the input for the remaining rounds. If not, undefined behaviour may arise and it won't be sure whether your program will be accepted or rejected. Also because of this, please be careful when hacking others' codes. In the sample, Chtholly won the game after the 3rd round, so it is required that your program doesn't read the number of the remaining 4th round.

The input format for hacking:

- The first line contains 3 integers n, m and c;
- The following m lines each contains an integer between 1 and c, indicating the number given to Chtholly in each round.

Problem D Connections

Time limit: 3 seconds

Hard times are coming to Byteland. Quantum computing is becoming mainstream and Qubitland is going to occupy Byteland. The main problem is that Byteland does not have enough money for this war, so the King of Byteland Byteman 0x0B had decided to reform its road system to reduce expenses.

Byteland has n cities that are connected by m one-way roads and it is possible to get from any city to any other city using these roads. No two roads intersect outside of the cities and no other roads exist. By the way, roads are one-way because every road has a halfway barrier that may be passed in one direction only. These barriers are intended to force enemies to waste their time if they choose the wrong way.

The idea of the upcoming road reform is to abandon some roads so that exactly $2n$ roads remain. Advisers of the King think that it should be enough to keep the ability to get from any city to any other city. (Maybe even less is enough? They do not know for sure.) The problem is how to choose roads to abandon. Everyone in Byteland knows that you are the only one who can solve this problem.

Input

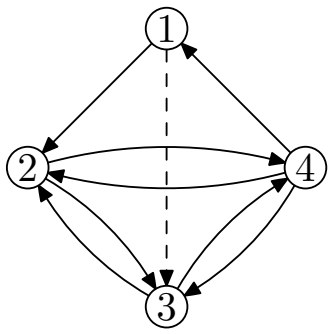
Input consists of several test cases. The first line of the input contains the number of tests cases.

The first line of each test case contains n and m — the number of cities and the number of roads correspondingly ($n \geq 4$, $m > 2n$). Each of the next m lines contains two numbers x_i and y_i denoting a road from city x_i to city y_i ($1 \leq x_i, y_i \leq n$, $x_i \neq y_i$). It is guaranteed that it is possible to get from any city to any other city using existing roads only. For each pair (x, y) of cities there is at most one road going from city x to city y and at most one road going from city y to city x. The solution is guaranteed to exist. The sum of m over all test cases in a single input does not exceed 100 000.

Output

For each test case output $m - 2n$ lines. Each line describes a road that should be abandoned. Print the road in the same format as in the input: the number of the source city and the number of the destination city. The order of roads in the output does not matter, but each road from the input may appear in the output at most once and each road in the output must have been in the input. It still must be possible to get from any city to any other city using the remaining roads.

Example

input	output	illustration
1 4 9 1 2 1 3 2 3 2 4 3 2 3 4 4 1 4 2 4 3	1 3	

Problem E Archery Tournament

Time limit: 3 seconds

You were invited to the annual archery tournament. You are going to compete against the best archers from all of the Northern Eurasia. This year, a new type of competition is introduced, where a shooting range is dynamic and new targets might appear at any second.

As the shooting range is far enough from you, it can be represented as a 2D plane, where $y = 0$ is the ground level. There are some targets in a shape of a circle, and all the targets are standing on the ground. That means, if a target's center is (x, y) $(y > 0)$, then its radius is equal to y, so that it touches the line $y = 0$. No two targets simultaneously present at the range at any given time intersect (but they may touch).

Initially, the shooting range is empty. Your participation in this competition can be described as n events: either a new target appears at the range, or you shoot an arrow at some point at the range. To hit a target, you must shoot strictly inside the circle (hitting the border does not count). If you shoot and hit some target, then the target is removed from the range and you are awarded one point.

Input

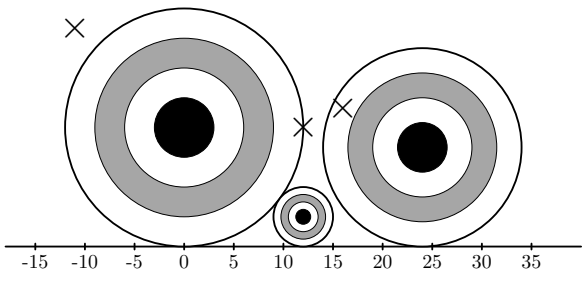
The first line of the input contains integer n $(1 \le n \le 2 \cdot 10^5)$. Next n lines describe the events happening at the tournament. The i-th line contains three integers t_i, x_i, and y_i $(t_i = 1, 2;\ -10^9 \le x_i, y_i \le 10^9;\ y_i > 0)$.

- If $t_i = 1$, then a new target with center (x_i, y_i) and radius y_i appears at the range.
- If $t_i = 2$, then you perform a shot, which hits the range at (x_i, y_i).

Output

For each of your shots, output a separate line with the single integer. If the shot did not hit any target, print "-1". If the shot hit a target, print the number of event when that target was added to the range. Events are numbered starting from 1.

Examples

input	output	illustration
8 1 0 12 2 -11 22 1 24 10 1 12 3 2 12 12 2 16 14 1 28 15 2 3 6	-1 -1 3 1	

Note

Illustration shows the state of the range after first six events. The rightmost target was hit by the last shot and is going to be removed.