

Problem A. Cableways

Input file: `input.txt`
Output file: `output.txt`
Time limit: 2 seconds
Memory limit: 64 mebibytes

Vova bought a new wide-angle lens for his camera a few days ago. Now he wants to take a panoramic photo with this lens, and that is the reason why he decided to climb the nearest hill.

There is a single narrow serpentine path leading from base of the hill to its top. Vova introduced coordinate axes in such a way that OY is directed along the hill side from base to top and OX is orthogonal to it. The path is represented as a polyline with n segments, vertices of which follow in the order of strictly increasing y -coordinate. For each segment of the path, Vova knows the time required to change x -coordinate by one while moving along this segment.

Moreover, Vova can use cableways. They start at some point of the path, run strictly up the hill parallel to OY axis and end at the next intersection with the path. Vova knows the time required to use each cableway.

Help Vova climb the hill before it gets dark! Note that Vova is so determined that he will refuse to go down the hill even if this action helps him to climb the hill faster.

Input

The first line contains a number of segments in the path n ($1 \leq n \leq 10^5$). The next line contains $n + 1$ integers which are x -coordinates of polyline vertices in the order from base to top. No two consecutive vertices have the same x -coordinate. The i -th of the following n lines contains integers v_i and m_i which are time required to change x -coordinate by one while moving along the i -th segment and a number of cableways starting at the i -th segment, respectively. The rest of this line contains m_i pairs of integers describing the cableways. Each cableway is described with x -coordinate of its start point and time required to use this cableway. The cableways are described in the order from base to top. No two cableways start at the same point. If a cableway starts at the common point of two path segments, it encounters in the description of only one segment. It is guaranteed that each cableway ends somewhere on the path. All numbers in lines are separated by spaces. All coordinates don't exceed 10^6 in their absolute value. All times are positive and don't exceed 10^6 . The total number of cableways doesn't exceed 10^5 .

Output

Output the minimal time required to climb the hill.

Examples

<code>input.txt</code>	<code>output.txt</code>
4 0 5 15 10 0 1 1 1 100 1 1 7 1 1 0 1 0	15
3 0 10 0 10 10 0 10 1 10 1 10 0	101

Problem B. Local Roots

Input file: .txt
Output file: output.txt
Time limit: 2 seconds
Memory limit: 64 mebibytes

Consider a word w consisting of n symbols. We can *decompose* it at point i ($1 \leq i \leq n - 1$) into a prefix p of length i and a suffix s of length $n - i$. *Local root* of a word w at point i is a non-empty word r such that:

- p is a suffix of r , or r is a suffix of p , or r is equal to p ;
- s is a prefix of r , or r is a prefix of s , or r is equal to s ;
- r has minimal possible length.

Your goal is to find such a point that the length of local root at this point is maximal possible.

Input

The only line contains a word w consisting of lowercase English letters. Its length is at least two and at most 300 000 symbols.

Output

Output the required point of decomposition and the length of local root at this point. If there are several possible answers, output any of them.

Example

input.txt	output.txt
aababaaa	5 6

Local roots of a word aababaaa at different points:

Point	Local root
1	a
2	babaa
3	ab
4	ba
5	aaabab
6	a
7	a

Problem C. Cutting a Parallelepiped

Input file: `input.txt`
Output file: `output.txt`
Time limit: 2 seconds
Memory limit: 64 mebibytes

Erno's small brother have lost his Rubik's cube. The next day he brought a saw. His goal was Erno's rectangular parallelepiped, all six faces of which were painted in different colors...

When Erno saw what was happening, the brother already made a few cuts parallel to the faces of the parallelepiped. The cuts passed through the whole parallelepiped. After all the cuts were made, the parallelepiped fell to pieces. Help Erno determine the initial position of each piece in the parallelepiped.

Input

The first line describes the original parallelepiped. It contains three space-separated integers, which are distance between front and rear faces d , distance between bottom and top faces h and distance between left and right faces w ($1 \leq d, h, w \leq 1000$). Then follows a string of six characters denoting colors of the faces. The colors are denoted by symbols «R», «O», «Y», «G», «B» and «V». The front bottom left vertex has coordinates $(0, 0, 0)$ and the rear top right vertex has coordinates (d, h, w) . The colors of the faces are described in the order: front, rear, bottom, top, left, right. The second line contains a number of pieces n ($1 \leq n \leq 1000$). The following n lines describe the pieces in the same format as the original parallelepiped. If a face of a piece was not a part of a face of the original parallelepiped, its color is denoted by «.».

Output

Output the position of each piece in the original parallelepiped. For each piece, output a symbol denoting a face which should become the front face, followed by a symbol denoting a face which should become the bottom face, followed by coordinates of the front bottom left vertex of this piece in the original parallelepiped. Denote the front face by «F», rear face by «B», bottom face by «D», top face by «U», left face by «L», right face by «R». The pieces should be described in the order of their appearance in the input. If there are several possible answers, output any of them. It is guaranteed that the solution exists.

Example

input.txt	output.txt
2 2 2 ROYGBV	F D 0 0 0
2	R F 1 0 0
1 2 2 R.YGBV	
2 2 1 YGVBO.	

Problem D. Warlord of the Army of Mages

Input file: .txt
Output file: output.txt
Time limit: 3 seconds
Memory limit: 64 mebibytes

After government of the Kingdom of Mages had formed a new army, they had to elect its warlord. Two wisest Masters, head of the Kingdom Zagamius and lich Sandro, decided to interview some candidates. They entered two different cabinets and started to invite certain mages for an interview or let go the mages they already talked to. Sandro and Zagamius could call the same mage for an interview several times and there could be more than one mage in their cabinets at some moments.

Each Master defined an order in which he will invite mages and let them go. If at certain moment both Masters can perform an action, then one of them performs it and the other one waits for him. In some cases only one Master can perform his action. For example, if Sandro invites a mage a , and this mage is being interviewed by Zagamius at that moment, then Sandro will wait until the mage a is free.

Unfortunately, there can be a situation when both Masters will wait for each other. In that case they won't elect the warlord of the army of mages and will lose a war that haven't yet started. You should determine whether such a stupid defeat is possible.

Input

The first line contains the number of test cases t ($1 \leq t \leq 10$). Each test case starts with a line containing space-separated integers n , m and k ($1 \leq n, m, k \leq 1000$), which are the number of actions planned by Sandro, the number of actions planned by Zagamius and the number of mages in the new army. Mages are numbered with integers from 1 to k . The next n lines contain the Sandro's plan. Each line contains a sign «+» or «-» and an integer i ($1 \leq i \leq k$). Sign «+» means that Sandro wants to invite the i -th mage, and sign «-» means that he wants to let him go. The next m lines contain the plan of Zagamius in the same format. In both plans Masters let a mage go only if he is present in his cabinet. It is guaranteed that no Master will invite a mage if he is already present in his cabinet. If all the planned actions are performed, the cabinets of both Masters will be empty.

Output

For each test case, output «:-)» if both Masters will certainly perform all the planned actions and will elect the warlord. In the other case, output «:-(».

Problem E. Integer-valued Complex Determinant

Input file: `input.txt`
Output file: `output.txt`
Time limit: 3 seconds
Memory limit: 64 mebibytes

Let $\mathbb{Z}[i]$ be a set of Gaussian integers, that is, a set of complex numbers with integer components. Let $a, b, q, r \in \mathbb{Z}[i]$, $a = bq + r$, $|r| < |b|$. Then r is a *remainder* of division of a by b .

Let $p \in \mathbb{Z}[i]$ and $X = (x_{ij})$ be a matrix of size $n \times n$ with elements from $\mathbb{Z}[i]$. Your goal is to calculate the remainder of division of determinant of X by p . Remember, that the determinant of X is equal to

$$\sum_{\pi} \text{sign}(\pi) \cdot x_{1\pi(1)} \cdot x_{2\pi(2)} \cdot \dots \cdot x_{n\pi(n)},$$

where the summation is taken over the set of all permutations π of n elements. Here addition and multiplication are the usual addition and multiplication of complex numbers.

Input

The first line contains an integer n ($1 \leq n \leq 50$). Each of the next n lines contains n space-separated complex numbers which are the elements of X . The last line contains a non-zero complex number p . Complex number is denoted by its real and imaginary parts, separated with space. All components of all complex numbers don't exceed 10^4 in their absolute value.

Output

Output real and imaginary parts of the remainder of division of determinant of X by p . If there are several possible answers, output any of them. It is guaranteed that the answer exists.

Example

<code>input.txt</code>	<code>output.txt</code>
2 2 0 -7 0 1 0 0 -1 3 1	3 0

Problem F. GCD 2010

Input file:
Output file:
Time limit: 2 seconds
Memory limit: 64 mebibytes

You have got a job offer from a secret project of the Agency of Federal Security under the code name "GCD 2010". The subject of research is a collection of positive integer numbers. Your goal is to calculate how the greatest common divisor of all numbers in this collection changes as we insert numbers into this collection and remove them from it. At the beginning of the experiment, the collection is empty.

Input

The first line contains an integer q ($1 \leq q \leq 10^5$), which is the number of operations with the collection. Each of the next q lines has either the form «+ x » or «- x ». In the first case, number x is inserted into the collection, in the latter case it is removed from the collection. The number x is a positive integer not exceeding 10^9 . It is guaranteed that operations remove only the integers which lie in the collection.

Output

Output the greatest common divisor of all numbers in the collection after each of the given operation. According to the 190R order, the greatest common divisor of an empty collection is equal to one.

Example

input.txt	output.txt
5	8
+ 8	2
+ 6	2
+ 8	2
- 8	6
- 8	

Problem G. Zamkadye

Input file: `input.txt`
Output file: `output.txt`
Time limit: 2 seconds
Memory limit: 64 mebibytes

There are n cities in the kingdom of Zamkadye, situated on the Great Plain. Once upon a time, a founder of the kingdom built $n - 1$ bidirectional roads. It is possible to get from any city to any other city using these roads. The founder was very avaricious, so the total length of the roads is minimal possible.

Karl the First, the king of Zamkadye, grew old and decided to start on a journey and visit all cities of his kingdom. He began his trip at the city of Ponaekhovsk, where a royal palace was situated, and acted in the following way:

- If Karl the First entered the city where he hadn't been before, he described his impressions in his diary and remembered the city, from which he came into this one.
- If there was a road to the city where Karl the First hadn't been before, he took this road. In the other case, Karl the First returned to the city, from which he came into this one for the first time.

Soon after the journey Karl the First died and his diary went to his son, Karl the Second. . .

A few decades later Karl the Second grew old and decided to visit all cities of his kingdom in the order they are described in the diary of his father. Karl the Second decided to use a zeppelin, which doesn't care about the roads and can fly from any city to any other city directly. Karl the Second was much lazier than his father, so he decided to visit each city exactly once and return to Ponaekhovsk after that.

A wife of Karl the Second asked for your help. She wants to know when her husband returns, but she doesn't know the order in which the cities are described in the diary. Calculate the minimal possible distance Karl the Second will have to cover during his journey.

Input

The first line contains a number of cities in Zamkadye n ($2 \leq n \leq 1000$). The cities are numbered with integers from 1 to n , Ponaekhovsk has number 1. The i -th of the next n lines contains coordinates of the i -th city. Coordinates are integers and don't exceed 10^4 in their absolute value. Then follows a blank line and $n - 1$ lines describing the roads in Zamkadye. Each road is described as a pair of numbers of cities it connects.

Output

Output the minimal distance Karl the Second will have to cover on zeppelin, with relative error not exceeding 10^{-9} .

Example

<code>input.txt</code>	<code>output.txt</code>
4 0 0 1 0 1 1 0 1 1 2 1 4 4 3	4.000

Problem H. Fly Hunt

Input file: `input.txt`
Output file: `output.txt`
Time limit: 6 seconds
Memory limit: 64 mebibytes

Pakhom and his friends decided to hunt for flies. Each of them took his favourite fly swatter. They crouched to a wall with n careless flies sitting on it and simultaneously swatted the wall so that each swatter left a mark on it. The mark of each swatter is a simple polygon with interior. No two swatter marks have common points. A fly was killed if it was situated inside or on the border of a swatter mark. Help friends calculate a number of flies killed by each of them.

Input

The first line contains a number of flies n ($1 \leq n \leq 10^5$). The next n lines contain coordinates of the flies. No two flies are situated at the same point. The next line contains a number of fly swatters m ($1 \leq m \leq 30\,000$). Each of the next m lines describes a mark of a fly swatter. A mark is described by a number of vertices in a corresponding polygon and coordinates of these vertices in counter-clockwise order. The total number of vertices in all polygons doesn't exceed 10^5 . All coordinates are integers and don't exceed 10^7 in their absolute value. All numbers in lines are separated by single spaces.

Output

Output the number of flies killed by each fly swatter. Describe the fly swatters in the order they are given in the input.

Example

<code>input.txt</code>	<code>output.txt</code>
3	0
0 0	2
1 1	1
4 4	
3	
3 -2 -2 -3 -3 -2 -3	
3 0 0 2 0 1 1	
3 3 0 5 0 4 6	

Problem I. Rabbit Hunt 2

Input file: `input.txt`
Output file: `output.txt`
Time limit: 8 seconds
Memory limit: 64 mebibytes

A good hunter kills two rabbits with one shot. But you need to kill the maximal possible number of rabbits with one shot to become the best hunter in the world. The program *Rabbit Hunt 2* might be quite helpful. This program treats all rabbits as points on a plane. It reads coordinates of n rabbits from the input, generates q different shooting propositions and chooses the one which leads to killing the maximal number of rabbits. A shooting proposition is defined by coordinates of the hunter and the direction of shooting. The shot kills all the rabbits lying on the corresponding ray including the rabbit situated at the same point as the hunter.

A proposition generator uses numbers $a_x, b_x, a_y, b_y, a_{vx}, b_{vx}, a_{vy}, b_{vy}$. Coordinates of the hunter in the first proposition is (x_1, y_1) , and a direction of shooting is (vx_1, vy_1) . Coordinates and direction in the i -th proposition ($i > 1$) are calculated using these formulas:

$$\begin{aligned}x_i &= (a_x * (x_{i-1} + 10\,000) + b_x) \bmod 20\,001 - 10\,000, \\y_i &= (a_y * (y_{i-1} + 10\,000) + b_y) \bmod 20\,001 - 10\,000, \\vx_i &= (a_{vx} * (vx_{i-1} + 10) + b_{vx}) \bmod 21 - 10, \\vy_i &= (a_{vy} * (vy_{i-1} + 10) + b_{vy}) \bmod 21 - 10.\end{aligned}$$

Unfortunately, the program *Rabbit Hunt 2* doesn't exist yet. Your goal is to create it.

Input

The first line contains a number of rabbits n ($1 \leq n \leq 10^4$). The next n lines contain the coordinates of rabbits. Coordinates are integers not exceeding 10^4 in their absolute value. No two rabbits are situated at the same point. The next line contains a number of shooting propositions q ($1 \leq q \leq 10^6$). The next line contains integers x_1, y_1, vx_1, vy_1 ($-10^4 \leq x_1, y_1 \leq 10^4$; $-10 \leq vx_1, vy_1 \leq 10$). The last line contains non-negative integers $a_x, b_x, a_y, b_y, a_{vx}, b_{vx}, a_{vy}, b_{vy}$. All these numbers don't exceed 10^5 . It is guaranteed that the direction of shooting is a non-zero vector for all q propositions. All numbers in lines are separated by single spaces.

Output

Output the number of proposition resulting in the maximal quantity of killed rabbits and this quantity. If there are several such propositions, output the one with the maximal number.

Example

input.txt	output.txt
4 0 0 2 0 3 0 2 2 3 -1 -1 1 0 1 0 1 1 1 0 1 0	2 3

Problem J. Space Tourism

Input file: Output file: output .txt
Time limit: 3 seconds
Memory limit: 64 mebibytes

There are n planets in Betelgeuse star system. Each planet is inhabited by a single race, different planets can be inhabited by the same race. All planets move in circular orbits, lie on a single ray starting at the star and have same constant angular velocity. Tourists from other galaxies often ask the Ministry of Space Tourism to help them plan their journey. They would like to be escorted to a planet, from which they can start their individual journey on their own small spaceship.

Spaceships of all the tourists are equipped with a small fuel tank, which allows to cover at most d astronomical units without reloading. Such a reloading can be made on any planet of the system. Moreover, each ship can operate only in a fixed range of distances to the star, where he can use the radiation energy. A tourist can be escorted to a planet only if the distance from this planet to the star fall into this range.

The Ministry ordered you to calculate the maximal number of races each tourist can come in contact with.

Input

The first line contains space-separated integers n , s and d ($1 \leq n, s, d \leq 10^5$), which are the number of planets in Betelgeuse star system, the number of known races and the maximal distance each ship can fly without reloading, respectively. Each of the next n lines describes a planet and contains two space-separated positive integers which are a distance from this planet to the star and a number of race which lives on this planet, respectively. Races are numbered with integers from 1 to s . The distances from planets to Betelgeuse are all different and don't exceed 10^6 astronomical units. The next line contains a number of tourists m ($1 \leq m \leq 10^5$). Each of the next m lines contains two space-separated positive integers, which are the minimal and the maximal possible distance to the star at which the ship of the tourist can operate in. These distances doesn't exceed 10^6 .

Output

For each tourist, you should output the maximal number of races he can come in contact with.

Example

input.txt	output.txt
5 5 11	2
10 1	1
50 2	
30 1	
20 1	
60 3	
2	
5 65	
15 45	