Problem Bull. Jupiter's Championship

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

As you probably know, in June Flatland will host the Jupiter's football championship. There will be n teams participating in the competition, and each team is going to play with each other team exactly once.

There are m different colors of football t-shirts produced on Jupiter. Each team will bring t-shirts of two **different** colors. Of course, during the single match all members of one team must wear t-shirts of the same colors, and this color should be different from the color of t-shirts opposite team is using in this game.

The president of the Fantastic Interplanet Football Association Joseph was invited to referee such an important tournament. At the beginning of each game Joseph can specify for each of the opposing teams the particular color of the t-shirts they should use during this game. He is able to choose colors only among the two colors this team has brought to the tournament. After this, Joseph chooses the color he is going to wear during the match. As you can guess, this color should differ from colors teams are using, so they are able to distinguish the referee and each team on the field. Any t-shirts may be used multiple times.

Joseph is going to buy the necessary t-shirts right before the start of the tournament, when he already knows for each team the colors of t-shirts this team has. Joseph cares a lot about the money of the federation, so he wants to buy the minimum number of t-shirts that is necessary to referee all the games. This problem is quite complicated for the common football referee, so he asks you to help.

Input

The first line of the input contains two integers n and m ($2 \le n \le 100\,000, 2 \le m \le 10^9$) — the number of teams participating in the championship and the number of different colors of football t-shirts produced on Jupiter.

The i-th of the following n lines contains two distinct integers from 1 to m — colors of the t-shirts that are available for the team number i.

Output

First print the minimum number of t-shirts required to be able to referee matches between all pairs of teams. If there is no solution, print `-1` instead.

Then print the colors of the t-shirts that form optimal answer.

If there are multiple possible solutions, you may print any of them.

Examples

standard input	standard output

`3 4`
`1 2`
`2 3`
`1 4`
	`1`
	`1`

`5 3`
`1 2`
`2 3`
`2 3`
`3 1`
`1 2`
	`2`
	`3 1`

Note

In the first sample it's enough to buy only t-shirt of color 1, cause in any match teams will be able to choose different colors, without using color 1. For example, in the game between first and second teams Joseph may tell team number 1 to use color 2 and team number 2 to use color 3.

Scoring

Tests for this problem are divided into three groups. For each group you earn points only if your solution passes all tests in this group and all tests in all **previous** groups. **Offline evaluation** means that your submission will be evaluated on the tests of the group only after the end of the contest.

Group	Tests	Points	Additional constraints		Comments
			n	m	
0	1–2	0	–	–	Sample tests
1	3–17	20	$2 \le n \le 10$	$2 \le m \le 10$	
2	18–32	20	$2 \le n \le 100$	$2 \le m \le 100$	
3	33–47	20	$2 \le n \le 10\,000$	$2 \le m \le 10^9$	
4	–	40	–	–	**Offline evaluation**

Problem Crocodile. Batman Returns

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2.5 seconds
Memory limit:	256 megabytes

Gotham City consists of a single street, and there are n skyscrapers located along it. They are numbered from west to east with integers from 1 to n, the height of the i-th skyscraper is equal to h_i meters.

Every night Batman performs an observation flight over the city. He climbs on the roof of some skyscraper and glides down to the roof of some other skyscraper. Due to the strong permanent wind he is only able to flight westward, but his altitude remains almost the same. Thus, he is able to glide down from skyscraper q to skyscraper p if and only if $p < q$ and $h_p < h_q$. Moreover, Batman is very manoeuvrable, so the height of the buildings between p and q don't matter. Batman cares a lot about the crime level in the city so he chooses such pair of valid p and q that $q - p$ is maximum possible.

City authorities have developed m plans of city renewal. According to the i-th plan only skyscrapers from l_i to r_i, inclusive will remain on this street, while others will be destroyed. For each plan i Batman wants to know the optimal plan to observe the city, namely such p_i and q_i that $l_i \le p_i < q_i \le r_i$, $h_{p_i} < h_{q_i}$ and $q_i - p_i$ is maximum possible.

Input

The first line of the input contains one integer n ($1 \le n \le 200\,000$) — number of skyscrapers on the street.

The second line contains n integers $h_i (1 \le h_i \le 200\,000)$ — heights of the skyscrapers.

Third line contains integer m ($1 \le m \le 200\,000$) — the number of plans designed by the city authorities.

Each of the last m lines contains two integers l_i and r_i ($1 \le l_i < r_i \le n$), denoting the range of the skyscrapers that will remain according to the i-th plan.

Output

For each renewal plan you should print two integers — optimal p_i and q_i. If there is no possible observation flight at all, you should print `-1 -1`.

If there are many optimal answers, you may print any one of them.

Examples

standard input	standard output
4 2 3 1 4 2 2 3 1 3	-1 -1 1 2
7 5 4 2 4 3 1 5 4 2 6 2 7 1 7 3 7	3 5 2 7 2 7 3 7

Note

Consider the first sample test. In the first query the only two available skyscrapers have heights 3 and 1 but they are not valid since $3 \geq 1$. In the second query the pair consisting of the first and the second skyscrapers is valid since they have heights 2 and 3.

Consider the second sample test. In the first query the pair of skyscrapers with heights 2 and 3, and the pair of skyscrapers with heights 2 and 4 are valid. The distance between first two of them is greater so this pair produces the answer for this query.

Scoring

Tests for this problem are divided into eleven groups. For each group you earn points only if your solution passes all tests in this group and all tests in all **previous** groups. **Offline evaluation** means that your submission will be evaluated on the tests of the group only after the end of the contest.

Group	Tests	Points	Additional constraints	Comments
			n, m	
0	1 – 2	0	–	Sample tests
1	3 – 12	10	$n, m \leq 100$	
2	13 – 22	10	$n, m \leq 500$	
3	23 – 32	10	$n, m \leq 1\,000$	
4	33 – 42	10	$n, m \leq 2\,000$	
5	43 – 52	10	$n, m \leq 5\,000$	
6	53 – 62	10	$n, m \leq 10\,000$	
7	63 – 72	10	$n, m \leq 20\,000$	
8	73 – 82	10	$n, m \leq 50\,000$	
9	–	10	$n, m \leq 100\,000$	**Offline evaluation**
10	–	10	$n, m \leq 200\,000$	**Offline evaluation**

Problem Eagle. The Evil League of Evil

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

Bad Horse is recruiting to The Evil League of Evil! He used his hoof to write down a long string s, consisting of letters "'texttt(", "'texttt)" and "`?`", and sent it to all applicants. Each person willing to join Evil League has to replace all characters "`?`" with either opening bracket or closing bracket. The invitation to join The Evil League of Evil will be send to the person who's resulting string contains the longest possible **subsequence**, that is correct bracket sequence.

Subsequence of the string s is the string that can be obtained by removing some characters (possibly none) from s. For example, strings "'texttabc", "`ac`", "`bcc`" and "`abbcc`" are subsequences of "`abbcc`", while "`cb`" and "`ba`" are not. Note, that the empty string is a subsequence of any string.

The sequence of brackets is called *correct* if:

1. it's empty;
2. it's a correct sequence of brackets, enclosed in a pair of opening and closing brackets;
3. it's a concatenation of two correct sequences of brackets.

For example, the sequences "`()()`" and "`((()))()`" are correct, while "`)(()`", "`(((((`" and "`())`" are not.

Dr. Horrible was dreaming of joining Evil League of Evil for year, but his pacifism blocks him from doing bad things. He is also bad in solving problems and asks you to deal with the Horse's puzzle.

Input

The only line of the input contains the string s ($1 \leqslant |s| \leqslant 10\,000\,000$).

It's guaranteed that s consists of letters "`(`", "`)`" and "`?`" only.

Output

Print the solution to the Evil Horse's puzzle that guarantees Doctor Horrible will be invited to join The Evil League of Evil. That is, replace "`?`" with either "`(`" or "`)`", to maximize the length of maximum correct bracket subsequence of the string. If there are many optimal answers, you may print any of them.

Examples

standard input	standard output
`?)))?)`	`()))()`
`)(???)(`	`)((())(`

Note

In the first sample, the resulting string contains correct bracket subsequence of length 4: "`()()`".

In the second sample, the resulting string contains correct bracket subsequence of length 4: "`(())`".

Scoring

Tests for this problem are divided into three groups. For each group you earn points only if your solution passes all tests in this group and all tests in all **previous** groups. **Offline evaluation** means that your submission will be evaluated on the tests of this group only after the end of the contest.

Group	Tests	Points	Additional constraints	Comments
			n	
0	1 – 2	0	–	Samples
1	3 – 31	10	$n \le 20$	
2	32 – 45	20	$n \le 1000$	
3	46 – 58	20	$n \le 10\,000$	
4	59 – 70	20	$n \le 100\,000$	
5	–	30	$n \le 10\,000\,000$	**Offline evaluation**, but the result for one test is available

Problem Moose. See the Sights on the Flights

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

Dima is an architect. He is also a photographer. He spends his time on travelling around the world and making photos of cool buildings like Big Ben etc.

This time Dima went to Berland famous with its subway system. It consists of n lines, each of which is represented with a line on the map of the city. For any two lines there is a subway station in their intersection point, those station entrances are considered to be the notable pieces of architecture. Dima decided to take a photo of them.

In order to take the panoramic photo, he is going to use a helicopter flight. Helicopter may use one of the t routes. Each route is also represented with a line on the map of the city. Dima is able to make a photo from an arbitrary point of the route, though the smaller distance from his location to the station means the better photo and the larger number of likes he is going to receive in social networks. That's why Dima needs your help.

You are given n descriptions of the subway lines and t lines defining the helicopter routes. For each of the helicopter routes Dima asks you to find the distance to the closest subway station.

It is guaranteed that no two subway lines coincide, any two subway lines have a common point, any two routes have a common point and each route has exactly one common point with each subway line.

Input

In the first line of the input there are two integers n, t ($2 \le n \le 100\,000$, $1 \le t \le 20$) — the number of subway lines and the number of helicopter routes, respectively.

In each of the following n lines there are three integers a_i, b_i and c_i ($|a_i|, |b_i| \le 10\,000$, $a_i^2+b_i^2 > 0$, $|c_i| \le 10^8$) defining each of the subway lines. The corresponding line is defined by the equation $a_i \cdot x + b_i \cdot y + c_i = 0$.

In each of the following t lines there are three integers u_i, v_i, w_i ($|u_i|, |v_i| \le 10\,000$, $u_i^2+v_i^2 > 0$, $|w_i| \le 10^8$) defining each of the helicopter routes. Similarly, each route is defined with the equation $u_i \cdot x + v_i \cdot y + w_i = 0$.

Output

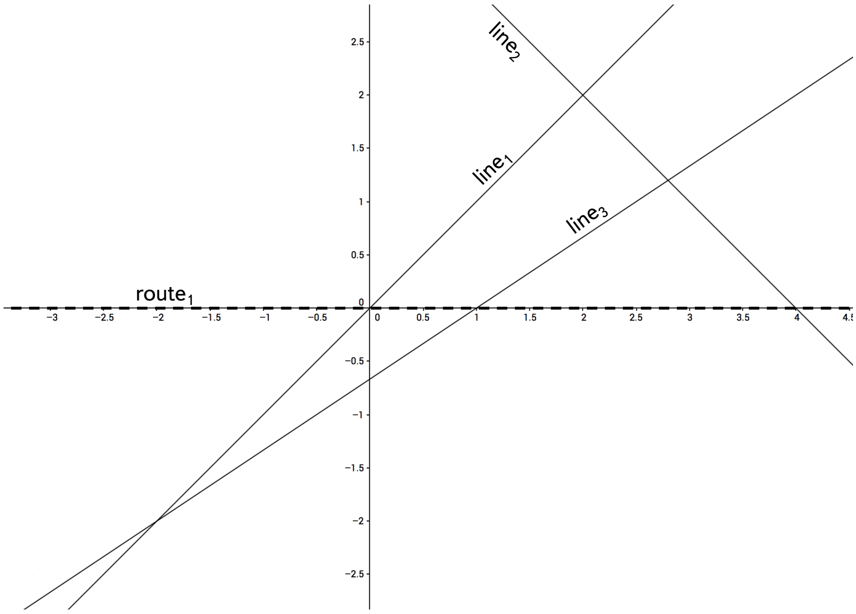
For each route output the only real number — the distance between i-th helicopter route and its most closest subway station. Your answer will be considered correct if the absolute or relative error between your answer and the answer of the jury doesn't exceed 10^{-9}. Namely, $\frac{|p-j|}{\max(1,j)} \le 10^{-9}$ where p is your answer and j is the answer of the jury.

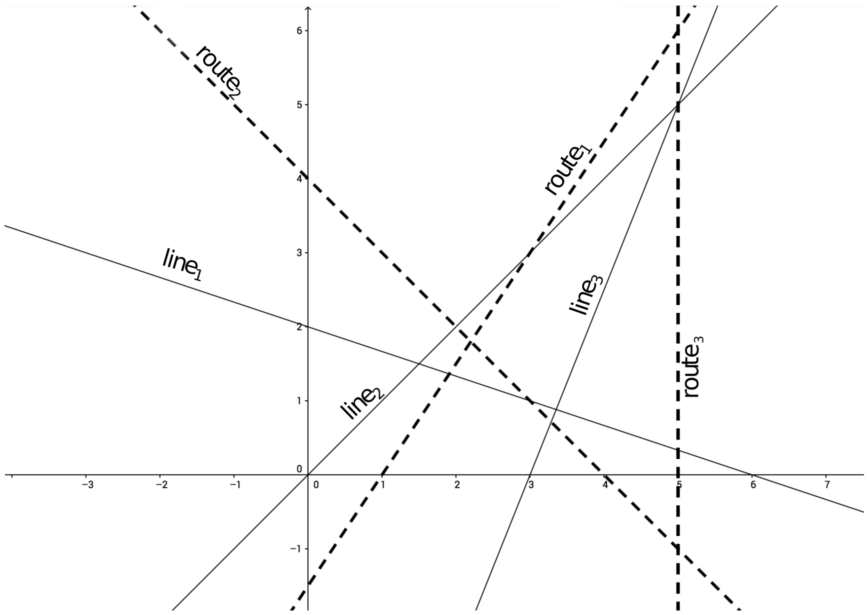
Examples

standard input	standard output
3 1 1 -1 0 1 1 -4 4 -6 -4 0 1 0	1.2
3 3 1 3 -6 -1 1 0 -5 2 15 3 -2 -3 -1 -1 4 1 0 -5	0.41602514717 0.16637806616 0.0

Note

The pictures for the samples are provided below.





Scoring

Tests for this problem are divided into five groups. For each group you earn points only if your solution passes all tests in this group and all tests in all **previous** groups except possibly the sample tests.

Group	Tests	Points	Constraints		Comment
			n	t	
0	1 – 2	0	–	–	Sample tests
1	3 – 19	10	$n \le 1000$	$t = 1$	$u_i = 0$
2	20 – 34	20	$n \le 1000$	$t = 1$	
3	35 – 55	30	$n \le 40\,000$	$t = 1$	
4	–	40	$n \le 100\,000$	$t \le 20$	**Offline evaluation**