Information

Source code limit

The size of each solution source code can't exceed 256 KiB.

Submissions limit

You can submit at most 50 solutions for each problem.

You can submit a solution to each task at most once per 30 seconds. This restriction does not apply in the last 15 minutes of the contest round.

Testing

Notice, that each subtask has a list of required subtasks. Subtask will be tested only if all tests of all required subtasks are passed. Be careful, some subtasks might not be tested, if your solution doesn't pass sample tests.

Scoring

We have two types of subtask scoring: "test" and "subtask".

"Test" means that points are awarded for each test in a subtask independently of other tests in this subtask.

"Subtask" means that points are awarded only if all tests in this subtask are passed.

For more information on subtask scoring read "Scoring" section of each problem.

The number of points scored for the problem is the total number of points scored on each of its subtasks. The score for the subtask is the maximum number of points earned for this subtask among all the solutions submitted.

Feedback

To get feedback for your solution, go to "Runs" tab in PCMS2 Web Client and use "View Feedback" link.

For more information, on what feedback is given, read "Feedback" section of each problem.

Problem A. File names

Time limit: 1 second
Memory limit: 512 megabytes

Petya and Vasya love to develop their own operating systems, so Petya uses PetOS operating system on his computer, and Vasya uses VasOS on his computer.

One day Petya wanted to send Vasya some files from his computer, but it turned out that it was not so easy to do. The problem is that in PetOS the name of file can be any string consisting of Latin letters and dots, with length in range from 1 to 20, and VasOS only supports the names of the form `<filename>.<extension>`, where `<filename>` and `<extension>` are non-empty strings of Latin letters, with the length of `<filename>` not greater than 8, and the length of `<extension>` not greater than 3.

Help Petya to calculate how many files from his list can be transferred to Vasya without changing their name.

Input

The first line contains the number n, the number of files Petya wants to transfer ($1 \le n \le 100$). The following n lines contain file names. All names have a length of 1 to 20 characters and contain only lowercase Latin letters and dots.

Output

Output the number of files that Petya can transfer to Vasya without renaming.

Scoring

All tests are graded independently.

Feedback

You will be able to see the outcome of your solution and the number of points scored on each test.

Example

standard input	standard output
8 .exe script a.in.txt a.t con.exe solution.cpp b.java verylongname.txt	3

Explanation

In the given example Petya can transfer the following files: `a.t`, `con.exe`, `solution.cpp`.

Problem B. Neatness

Time limit: 1 second
Memory limit: 512 megabytes

Boys Dima and Mitya share a room in a dormitory and take turns cleaning it every k days. New month, consisting of n days, has just begun, so boys have to come up with a new cleaning schedule. To do it, boys simply choose who is going to clean first, and which of the first k days of the month it will happen. The next cleaning will happen exactly k days after the previous one, and the other boy will have to do it. For example, if Dima cleans the room on day i, then, on day $i + k$, Mitya will have to do the cleaning, on day $i + 2k$, it will be Dima's turn again, and so on.

The schedule should be fair: both boys should clean the room equal number of times. The situation is complicated by the fact that both Dima and Mitya are planning go to an olympiad once during this month. If one of the boys is absent during his cleaning day, the other boy is cleaning in his place. The schedule does not shift in this case. It is known that Dima will be absent from the a-th day to the b-th day inclusive, and Mitya — from the c-th day until the d-th day. Days in month are numbered from 1 to n. Boys' trips to olympiads do not intersect, so every day at least one of the boys is at home.

Help boys decide who should clean the room first, and which day (among the first k days of the month) it should happen, or determine that it is impossible.

Input

First line contains two integers n and k — the number of days in this month and the number of days between two consecutive cleanings ($2 \le n \le 10^{18}$, $1 \le k \le n$).

Second line contains integers a, b, c and d, describing boys' trips to olympiads ($1 \le a \le b \le n$; $1 \le c \le d \le n$; $b < c$ or $d < a$).

Output

If it's impossible to create a fair schedule, print -1.

Otherwise, print one integer in the first line — the day of the first cleaning. On the second line print «`Dima`», if Dima should clean first, and «`Mitya`» otherwise.

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

Scoring

Subtask	Points	Constraints	Scoring	Required subtasks
1	17	$n \le 100$; $k \le 100$	subtask	—
2	24	$n \le 10^5$; $k \le 10^5$	subtask	1
3	33	$n \le 10^{18}$; $k \le 10^5$	subtask	1, 2
4	26	$n \le 10^{18}$; $k \le 10^{18}$	subtask	1, 2, 3

Feedback

In subtasks 1, 2, and 3 you will see the outcome of your solution on each test.

In subtask 4 you will see the number of points and the outcome on the first failed test.

Examples

standard input	standard output
12 5 6 10 11 12	3 Dima
18 4 15 18 1 6	3 Mitya
10 3 1 4 5 6	-1

Explanations

Let's represent sample tests as tables using the following notation:

«d» — day, when Dima is away,

«m» — day, when Mitya is away,

«D» — day, when Dima cleans the room,

«M» — day, when Mitya cleans the room,

«*» — possible starting day.

Then for the first sample we get the following table:

Days	1	2	3	4	5	6	7	8	9	10	11	12
Departures	*	*	*	*	*	d	d	d	d	d	m	m
Schedule	.	.	D	.	.	.	.	M	.	.	.	.
Actual	.	.	D	.	.	.	.	M	.	.	.	.

This way, both bots will clean the room equal number of times, and boys' trips do not affect the schedule.

Table for the second sample:

Days	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Departures	m*	m*	m*	m*	m	m	.	.	.	.	.	.	.	.	d	d	d	d
Schedule	.	.	M	.	.	.	D	.	.	.	M	.	.	.	D	.	.	.
Actual	.	.	D	.	.	.	D	.	.	.	M	.	.	.	M	.	.	.

Because Mitya is absent on the third day, Dima will clean instead. Mitya will clean the room on the 15-th day because of Dima's trip. This way, each boy cleans the room twice, even if it wasn't scheduled.

Problem C. Tax Collection

Time limit: 2 seconds
Memory limit: 512 megabytes

Byteland is a rectangle of $n \times m$, divided into $n \cdot m$ square provinces. Recently in Byteland tax reform was carried out, as a result of which the number $a[i, j]$ was fixed for each province. If $a[i, j] > 0$, then the province in the square (i, j) must pay $a[i, j]$ bytecoins every month to the budget. If $a[i, j] < 0$, then the province (i, j) is subsidized and receives $-a[i, j]$ bytecoins from the budget.

To collect taxes, the government has developed the following scheme. In one of the provinces a treasury will be built. Every month a tax collector will leave this building. He will go round all the provinces, collecting taxes and giving out subsidies, and then going back to the treasury. His path must satisfy the following properties:

- the path must begin in the province in which the treasury is located,
- the path must end in a province that has a common side with the province in which the treasury is located,
- each province should be visited **exactly** once,
- neighboring provinces along the path should have a common side.

The government wants to choose a province for the treasury and a path for the collector in such a way that for each subsidized province the collector can give them the right amount of money from the previously collected. Help them build such a path or say that it is impossible.

Input

The first line contains two integers n and m — the size of Byteland ($2 \le n, m \le 300$).

Each of next n rows consists of m integers $a[i, j]$. These lines describe the provinces: $a[i, j]$ is the value by which the number of bytecoins owned by the collector will change when visiting the province located at (i, j) ($1 \le |a[i, j]| \le 10\,000$).

Output

If solution exists, output $n \cdot m$ pairs of numbers, the coordinates of the provinces that the collector should visit in the order in which he should visit them. If there is no solution, output -1 .

Scoring

Subtask	Points	Constraints			Scoring	Req. subtasks
		n	m	Additional constraints		
1	23	$n = 2$	$m \le 300$	—	subtask	—
2	15	$n \le 300$	$m \le 300$	all $a_{i,j} > 0$	subtask	—
3	29	$n \le 300$	$m \le 300$	exactly one $a_{i,j} < 0$	subtask	—
4	33	$n \le 300$	$m \le 300$	—	subtask	1, 2, 3

Feedback

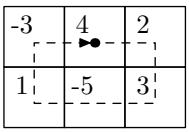
You will see the outcome of your solution on each test of each subtask.

Examples

standard input	standard output
2 3 -3 4 2 1 -5 3	1 2 1 3 2 3 2 2 2 1 1 1
4 4 1 -5 -3 1 1 5 2 -2 4 1 -3 1 -8 6 -2 3	2 3 2 2 1 2 1 1 2 1 3 1 4 1 4 2 3 2 3 3 4 3 4 4 3 4 2 4 1 4 1 3
2 2 1 -2 -1 1	-1
3 3 1 1 1 1 1 1 1 1 1	-1

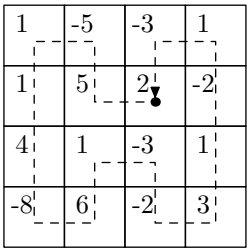
Note

Path for the first example:



Sum of bytecoins that collector has after each province: 4, 6, 9, 4, 5, 2.

Path for the second example:



Sum of bytecoins that collector has after each province: 2, 7, 2, 3, 4, 8, 0, 6, 7, 4, 2, 5, 6, 4, 5, 2.

Problem D. Road Building

Time limit: 2 seconds
Memory limit: 512 megabytes

The burger eating contest is coming! The contest is taking place at two different cities. But there was a slight miscalculation — there is no direct road between these cities across the forest. To save enough time organizers decided to build a direct road with maximum width through the forest without cutting any trees. Moreover, it was decided to leave at least one tree on both sides of the road.

For simplicity, let's represent each tree as a circle on plane with Cartesian coordinates. Build a road of infinite length and maximum width parallel to straight line, that passes through two given points.

Input

First line contains four integers x_s, y_s, x_f and y_f — coordinates of points (x_s, y_s) and (x_f, y_f) that the line passes through.

Second line contains single integer n — number of trees in the forest ($2 \le n \le 2 \cdot 10^5$). There are no intersecting trees, but they can touch each other.

Each of the next n lines contains three integers x_i, y_i and r_i — coordinates of the center and radius of i-th tree ($1 \le r_i \le 10^9$).

All coordinates' absolute values don't exceed 10^9.

Output

Output maximum width of the road parallel to straight line, that passes through points (x_s, y_s) and (x_f, y_f). There should be at least one tree on both sides of the road. The road may touch the trees, but not intersect them. If there is no such road, output 0.

Your answer will be accepted, if it's absolute or relative error doesn't exceed 10^{-6}. Formally speaking, if your answer is a and jury's is b, then it will be accepted if $\frac{|a-b|}{\max(1,b)} \le 10^{-6}$.

Scoring

Subtask	Score	Constraints	Scoring	Required subtasks
1	27	$2 \le n \le 100$	subtask	—
2	19	All radii are equal	subtask	—
3	23	The straight line is parallel to axes	subtask	—
4	31	No additional constraints	subtask	1, 2, 3

Feedback

In subtasks 1, 2, and 3 you will be able to see the outcome for each test.

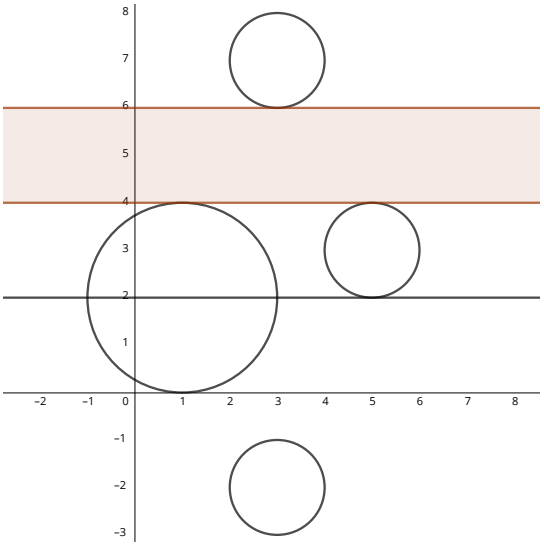
In subtask 4 you will be able to see the total score for the subtask and the outcome for the first failed test.

Examples

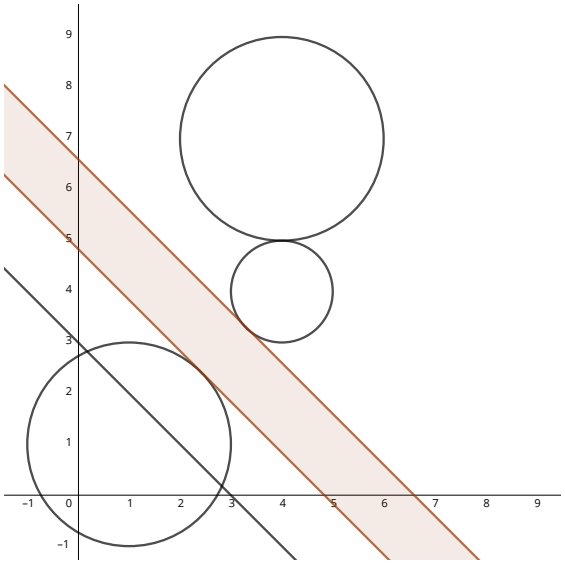
standard input	standard output
0 2 4 2 4 5 3 1 3 -2 1 1 2 2 3 7 1	2.000000000000000
0 3 3 0 3 1 1 2 4 4 1 4 7 2	1.242640687119285

Explanations

First example:



Second example:



Problem E. The Expert

Time limit: 5 seconds
Memory limit: 512 megabytes

> We need you to draw seven red lines, all of them strictly perpendicular; some with green ink and some with transparent, and one in the form of a kitten.
>
> Technical specification

You're asked to draw n distinct lines on a plane that are parallel to the coordinate axes. Some pairs of the lines are required to be parallel, and some pairs are required to be perpendicular. All lines must be described by equations $a \cdot x + b \cdot y + c = 0$, where a, b and c are integers. Let the i-th line be described as $a_i \cdot x + b_i \cdot y + c_i = 0$. Your task is to minimize the size of set containing all numbers a_i, b_i and c_i. In other words, you need to minimize the number of different coefficients used in description of all lines.

Calculate the minimum number of different coefficients used to draw the lines, or report that it is impossible. If there is a solution, find any way to do it using the smallest number of different coefficients.

Input

First line contains two integers n and m, number of lines and number of requirements ($1 \le n, m \le 10^6$).

Next m lines describe requirements. The i-th of these lines consists of three integers t_i, p_i and q_i: if t_i is equal to 0, then lines p_i and q_i must be parallel, otherwise, lines p_i and q_i must be perpendicular to each other ($t_i \in \{0, 1\}$; $1 \le p_i, q_i \le n$; $p_i \ne q_i$).

Output

If no solution exists, print -1.

If solution exists, print the minimal number of different coefficient. Each of the next n lines should consist of three integers, a_i, b_i and c_i — the i-th line coefficients. All coefficients mustn't exceed 10^9 by their absolute value.

Scoring

Subtask	Points	Constraints	Scoring	Required subtasks
1	10	$n, m \le 15$	subtask	—
2	20	$n \le 5\,000$, all $t_i = 0$	subtask	—
3	20	$m = n - 1$, $p_i = 1$, $q_i = i + 1$	subtask	—
4	20	$n \le 5\,000$	subtask	1, 2
5	15	$n \le 100\,000$	subtask	1, 2, 4
6	15	No additional constraint	subtask	1, 2, 3, 4, 5

Feedback

In all subtasks you will be able to see the total score for the subtask and the outcome for the first failed test.

Examples

standard input	standard output
3 3 1 3 2 0 1 3 1 1 2	2 -7 0 0 0 -7 -7 -7 0 -7
2 2 1 1 2 0 2 1	-1

Explanations

In the first example one of the ways to draw the lines:

- $-7 \cdot x + 0 \cdot y + 0 = 0$
- $0 \cdot x - 7 \cdot y - 7 = 0$
- $-7 \cdot x + 0 \cdot y - 7 = 0$

In the second example lines 1 and 2 has to be perpendicular and parallel at the same time. No such lines exist.