Problem A. Physical Laboratory

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

Chiaki has n bottles of cold water with temperature C and m bottles of hot water with temperature H. Each bottle has a certain volume.

Chiaki has done a series of experiments. Each time, she chose one bottle with hot and one with cold water, mixed their contents and measured the temperature, then poured the mixture to the two bottles with the same volume before the experiment.

Chiaki has recorded k temperature. The temperature is written in the form of a fraction p/q. We will assume that during the experiment there is no loss of energy. Then, when the liquids are mixed with the volumes U and V and the temperatures X and Y, the resulting mixture has a temperature $(UX + VY)/(U + V)$.

Chiaki would like to know whether her records are all correct.

Input

The first line contains 5 integers n, m, C, H and k ($1 \le n, m, C, H, k \le 10^5, C < H$).

The second line contains n integers $c_1, \ldots, c_n$ ($1 \le c_i \le 10^5$) – volumes of bottles with cold water.

The third line contains m integers $h_1, \ldots, h_m$ ($1 \le h_i \le 10^5$) – volumes of bottles with hot water.

The following k lines contain two integers p_i and q_i ($1 \le p_i, q_i \le 10^5$) – the numerator and denominator of the fraction, which was recorded in the i-th record.

Output

Output k lines. In the i-th line, output "YES"(without the quotes) if the record is correct, or "NO"(without the quotes) otherwise.

Example

standard input	standard output
3 2 30 50 4 2 3 5 5 10 40 1 45 1 140 3 170 4	YES NO YES YES

Problem B. Salary

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

There n employees in SUA. The salary of each employee of this company consists of two parts: a salary and a bonus. So, the i-th employee receives x_i rubles of salary and y_i rubles of bonus every month.

The management of the company became aware that its employees began to exchange information about their salaries. The management found out that there are m such pairs of employees u_i and v_i that the employee u_i regularly informs employee v_i his salary. As a punishment for violation of corporate etiquette, the management of the company wants to upset its employees as much as possible. Management believes that the goal is achieved if for each pair of employees u and v, such that u informs his salary to v, at least one of the following two conditions is fulfilled:

- the salary of employee v is less than employee u.
- the bonus of employee v is less than employee u.

In the process of achieving this requirement, management does not want to make significant changes in the company's salary policy. Therefore, the only thing that is ready to make is to select several employees and swap the amount of his bonus and the amount of his salary. In addition, the number of employees who will have to swap salary and bonus should be minimal.

Input

There are multiple test cases. The first line of input contains an integer T, indicating the number of test cases. For each test case:

The first line of contains 2 integers n and m – the number of employees in the company and the number of pairs of employees that management has caught in the exchange of information.

Each of the following n lines contains 2 integers x_i and y_i ($1 \le x_i, y_i \le 10^9$) – the salary and bonus of the corresponding employee.

Each of the following m lines contains 2 integers u_i and v_i ($1 \le u_i \ne v_i \le n$). Each such line means that the employee u_i informs his salary to the employee v_i.

It is guaranteed that neither the sum of all n nor the sum of all m will exceed 10^5.

Output

For each test case, output -1 in a separate line if the leadership requirements are not met. If they are doable, output the number k – the minimum number of employees who need to swap the amount of salary and bonus, and then k different numbers separated by spaces – the indices of these employees. In case there are several optimal answers, output any of them.

Examples

standard input	standard output
2 3 3 2 2 1 1 3 3 2 1 3 2 1 3 3 2 1 2 2 3 3 4 2 1 3 2	-1 0
1 3 2 1 2 1 2 3 4 2 1 3 2	1 1

Problem C. Robot

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	256 megabytes

On an infinite cellular field, there is a robot in the cell (x_1, y_1). After that, it performs exactly t moves to the neighboring cell (left, right, up and down) on the side and is in the cell (x_2, y_2).

It is known that in the process of moving the robot always had positive x and y coordinates. It is also known that the robot first appeared in the cell (x_2, y_2) after making the t-th move.

It is required to calculate the number of ways the robot travels, which are suitable for all the conditions described above. Since this number can be quite large, output it modulo 998 244 353. It is known that the initial cell of the robot does not coincide with the final one, and also that they have positive x and y coordinates.

Input

The first line contains five integers x_1, y_1, x_2, y_2 and t ($1 \le x_1, y_1, x_2, y_2, t \le 250000$). The initial and final cells do not coincide.

Output

Output the number of ways that the robot could get from one cell to another, modulo 998 244 353.

Examples

standard input	standard output
1 1 2 2 2	2
1 1 2 2 4	8
1 1 2 2 15	0

Problem D. Game

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

Two teams are playing an intellectual game. The first team has n players and the second team has m players. There are k clues in total. Each player know some of the clues.

During the game, players can exchange clues according to the following rules: Each player of the first team can approach one player of the second and ask him for a clue that he does not yet know. If there are several such clues, the player of the second team will inform him a random one.

Each player of the first team can ask for a hint only from one player of the second team, while several players can apply to the same player of second team. The first team wins if they manage to collect all the clues.

Help the captain of the first team find out if his team will be able to collect all the clues, regardless of the responses of the second team.

Input

The first line contains 3 integers n, m and k ($1 \le n, m \le 500, 1 \le k \le 5000$) – the number of players in each team and the number of clues, respectively.

Each of the following $n + m$ lines contains the clues available to players at the beginning of the game in the following format: the first number in the line corresponds to the number of clues, and the following numbers contain the clues – natural numbers not exceeding k.

Output

If the first team can collect all the clues, output 1 in the first line. Then output n integers in the second line where the i-th integer means the index of player in the second time that the i-th player in the first team should contact to.

If the first team cannot collect all the clues, just output 2 in a single line.

Examples

standard input	standard output
3 2 4 1 1 1 2 1 3 2 1 4 1 3	1 1 1 1
3 2 4 1 1 1 2 1 3 3 1 2 4 3 1 3 4	2

Problem E. Knapsack

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

Chiaki has an integer m. Consider the following knapsack problem: given N things with weights a_i, find the number of ways to choose a subset of things with a total weight equal to exactly W?

Chiaki would like to construct a knapsack problem whose answer is a multiple of m.

Input

The first contains an integer m ($1 \leq m \leq 10^{18}$).

Output

If there is no such test, print a single -1. Otherwise, in the first line output two integers N and W ($1 \leq N \leq 200, 1 \leq W \leq 500$). In the next line output N integers a_i ($1 \leq a_i \leq W$).

Example

standard input	standard output
9	10 500 111 500 500 500 500 500 500 500 500 500

Problem F. DSU

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

Consider the following pseudocode of *union* and *find* operation for *disjoint uinon sets*:

```
union(a, b)
  if rank[a] == rank[b] then
    rank[a] = rank[a] + 1
  if rank[a] > rank[b] then
    parent[b] = a
  else
    parent[a] = b

find(a)
  while parent[a] != a do
    a = parent[a]
  return a
```

Initially, all the value in *rank* array is zero and $parent[i] = i$.

You are given an array of $parent[i]$. Is there such a sequence of operations *union*, that after it is executed, the parent array becomes equal to the specified one?

Input

The first line contains an integer n ($1 \le n \le 10^4$).

The second line contains n integers $parent[1], \ldots, parent[n]$ ($1 \le parent[i] \le n$ for all i from 1 to n).

Output

If the desired sequence does not exist, print -1. Otherwise, in the first line print the integer $k \ge 0$ – the number of operations.

Then output k pairs of numbers a_i and b_i ($1 \le a_i, b_i \le n$) – arguments of the i-th operation *union*. By the time of the i-th operation, a_i and b_i must be the roots of different trees.

Examples

standard input	standard output
4 1 1 1 4	2 1 2 1 3
3 2 3 3	-1