Problem A. Linear Keyboard

Time Limit 1000 ms
Mem Limit 262144 kB

You are given a keyboard that consists of 26 keys. The keys are arranged sequentially in one row in a certain order. Each key corresponds to a unique lowercase Latin letter.

You have to type the word s on this keyboard. It also consists only of lowercase Latin letters.

To type a word, you need to type all its letters consecutively one by one. To type each letter you must position your hand exactly over the corresponding key and press it.

Moving the hand between the keys takes time which is equal to the absolute value of the difference between positions of these keys (the keys are numbered from left to right). No time is spent on pressing the keys and on placing your hand over the first letter of the word.

For example, consider a keyboard where the letters from '`a`' to '`z`' are arranged in consecutive alphabetical order. The letters '`h`', '`e`', '`l`' and '`o`' then are on the positions 8, 5, 12 and 15, respectively. Therefore, it will take $|5-8|+|12-5|+|12-12|+|15-12|=13$ units of time to type the word "`hello`".

Determine how long it will take to print the word s.

Input

The first line contains an integer t ($1 \le t \le 1000$) — the number of test cases.

The next $2t$ lines contain descriptions of the test cases.

The first line of a description contains a keyboard — a string of length 26, which consists only of lowercase Latin letters. Each of the letters from '`a`' to '`z`' appears exactly once on the keyboard.

The second line of the description contains the word s. The word has a length from 1 to 50 letters inclusive and consists of lowercase Latin letters.

Output

Print t lines, each line containing the answer to the corresponding test case. The answer to the test case is the minimal time it takes to type the word s on the given keyboard.

Examples

Input	Output
5 abcdefghijklmnopqrstuvwxyz hello abcdefghijklmnopqrstuvwxyz i abcdefghijklmnopqrstuvwxyz codeforces qwertyuiopasdfghjklzxcvbnm qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq qqqqqqqqq qwertyuiopasdfghjklzxcvbnm abacaba	13 0 68 0 74

Problem B. Dreamoon Likes Coloring

Time Limit 2000 ms
Mem Limit 262144 kB

Dreamoon likes coloring cells very much.

There is a row of n cells. Initially, all cells are empty (don't contain any color). Cells are numbered from 1 to n.

You are given an integer m and m integers $l_1, l_2, \ldots, l_m$ $(1 \le l_i \le n)$

Dreamoon will perform m operations.

In i-th operation, Dreamoon will choose a number p_i from range $[1, n - l_i + 1]$ (inclusive) and will paint all cells from p_i to $p_i + l_i - 1$ (inclusive) in i-th color. Note that cells may be colored more one than once, in this case, cell will have the color from the latest operation.

Dreamoon hopes that after these m operations, all colors will appear at least once and all cells will be colored. Please help Dreamoon to choose p_i in each operation to satisfy all constraints.

Input

The first line contains two integers n, m $(1 \le m \le n \le 100\,000)$.

The second line contains m integers $l_1, l_2, \ldots, l_m$ $(1 \le l_i \le n)$.

Output

If it's impossible to perform m operations to satisfy all constraints, print "'`-1`" (without quotes).

Otherwise, print m integers $p_1, p_2, \ldots, p_m$ $(1 \le p_i \le n - l_i + 1)$, after these m operations, all colors should appear at least once and all cells should be colored.

If there are several possible solutions, you can print any.

Examples

Input	Output
5 3 3 2 2	2 4 1

Input	Output
10 1 1	-1

Problem C. Elephant

Time Limit 1000 ms
Mem Limit 262144 kB

An elephant decided to visit his friend. It turned out that the elephant's house is located at point 0 and his friend's house is located at point $x(x > 0)$ of the coordinate line. In one step the elephant can move $1, 2, 3, 4$ or 5 positions forward. Determine, what is the minimum number of steps he need to make in order to get to his friend's house.

Input

The first line of the input contains an integer x ($1 \le x \le 1\,000\,000$) — The coordinate of the friend's house.

Output

Print the minimum number of steps that elephant needs to make to get from point 0 to point x.

Examples

Input	Output
5	1

Input	Output
12	3

Note

In the first sample the elephant needs to make one step of length 5 to reach the point x.

In the second sample the elephant can get to point x if he moves by 3, 5 and 4. There are other ways to get the optimal answer but the elephant cannot reach x in less than three moves.

Problem D. President and Roads

Time Limit 2000 ms
Mem Limit 262144 kB

Berland has n cities, the capital is located in city s, and the historic home town of the President is in city t ($s \neq t$). The cities are connected by one-way roads, the travel time for each of the road is a positive integer.

Once a year the President visited his historic home town t, for which his motorcade passes along some path from s to t (he always returns on a personal plane). Since the president is a very busy man, he always chooses the path from s to t, along which he will travel the fastest.

The ministry of Roads and Railways wants to learn for each of the road: whether the President will definitely pass through it during his travels, and if not, whether it is possible to repair it so that it would definitely be included in the shortest path from the capital to the historic home town of the President. Obviously, the road can not be repaired so that the travel time on it was less than one. The ministry of Berland, like any other, is interested in maintaining the budget, so it wants to know the minimum cost of repairing the road. Also, it is very fond of accuracy, so it repairs the roads so that the travel time on them is always a positive integer.

Input

The first lines contain four integers n, m, s and t ($2 \leq n \leq 10^5$; $1 \leq m \leq 10^5$; $1 \leq s, t \leq n$) — the number of cities and roads in Berland, the numbers of the capital and of the Presidents' home town ($s \neq t$).

Next m lines contain the roads. Each road is given as a group of three integers a_i, b_i, l_i ($1 \leq a_i, b_i \leq n$; $a_i \neq b_i$; $1 \leq l_i \leq 10^6$) — the cities that are connected by the i-th road and the time needed to ride along it. The road is directed from city a_i to city b_i.

The cities are numbered from 1 to n. Each pair of cities can have multiple roads between them. It is guaranteed that there is a path from s to t along the roads.

Output

Print m lines. The i-th line should contain information about the i-th road (the roads are numbered in the order of appearance in the input).

If the president will definitely ride along it during his travels, the line must contain a single word "`YES`" (without the quotes).

Otherwise, if the i-th road can be repaired so that the travel time on it remains positive and then president will definitely ride along it, print space-separated word "`CAN`" (without the quotes), and the minimum cost of repairing.

If **we can't make the road be such that president will definitely ride along it**, print "`NO`" (without the quotes).

Examples

Input	Output
6 7 1 6 1 2 2 1 3 10 2 3 7 2 4 8 3 5 3 4 5 2 5 6 1	YES CAN 2 CAN 1 CAN 1 CAN 1 CAN 1 YES

Input	Output
3 3 1 3 1 2 10 2 3 10 1 3 100	YES YES CAN 81

Input	Output
2 2 1 2 1 2 1 1 2 2	YES NO

Note

The cost of repairing the road is the difference between the time needed to ride along it before and after the repairing.

In the first sample president initially may choose one of the two following ways for a ride: $1 \to 2 \to 4 \to 5 \to 6$ or $1 \to 2 \to 3 \to 5 \to 6$.

Problem E. The Two Routes

Time Limit 2000 ms
Mem Limit 262144 kB

In Absurdistan, there are n towns (numbered 1 through n) and m bidirectional railways. There is also an absurdly simple road network — for each pair of different towns x and y, there is a bidirectional road between towns x and y **if and only if** there is no railway between them. Travelling to a different town using one railway or one road always takes exactly one hour.

A train and a bus leave town 1 at the same time. They both have the same destination, town n, and don't make any stops on the way (but they can wait in town n). The train can move only along railways and the bus can move only along roads.

You've been asked to plan out routes for the vehicles; each route can use any road/railway multiple times. One of the most important aspects to consider is safety — in order to avoid accidents at railway crossings, the train and the bus must not arrive at the same town (except town n) simultaneously.

Under these constraints, what is the minimum number of hours needed for both vehicles to reach town n (the maximum of arrival times of the bus and the train)? Note, that bus and train are not required to arrive to the town n at the same moment of time, but are allowed to do so.

Input

The first line of the input contains two integers n and m ($2 \le n \le 400$, $0 \le m \le n(n - 1) / 2$) — the number of towns and the number of railways respectively.

Each of the next m lines contains two integers u and v, denoting a railway between towns u and v ($1 \le u, v \le n$, $u \ne v$).

You may assume that there is at most one railway connecting any two towns.

Output

Output one integer — the smallest possible time of the later vehicle's arrival in town n. If it's impossible for at least one of the vehicles to reach town n, output -1.

Examples

Input	Output
4 2 1 3 3 4	2

Input	Output
4 6 1 2 1 3 1 4 2 3 2 4 3 4	-1

Input	Output
5 5 4 2 3 5 4 5 5 1 1 2	3

Note

In the first sample, the train can take the route $1 \rightarrow 3 \rightarrow 4$ and the bus can take the route $1 \rightarrow 2 \rightarrow 4$. Note that they can arrive at town 4 at the same time.

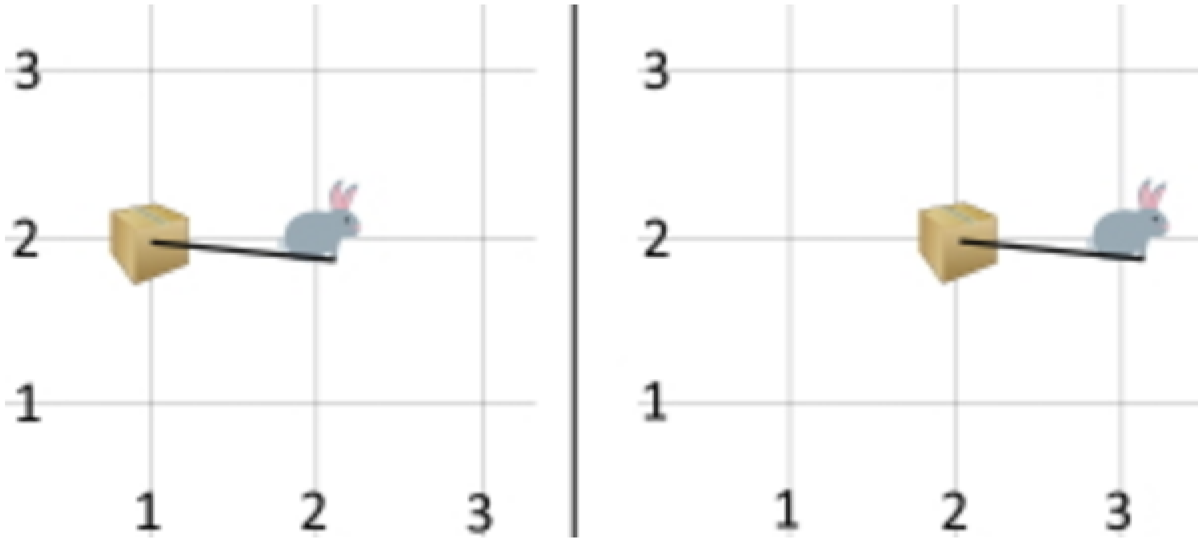
In the second sample, Absurdistan is ruled by railwaymen. There are no roads, so there's no way for the bus to reach town 4.

Problem F. Box is Pull

Time Limit 1000 ms
Mem Limit 262144 kB

Wabbit is trying to move a box containing food for the rest of the zoo in the coordinate plane from the point (x_1, y_1) to the point (x_2, y_2).

He has a rope, which he can use to pull the box. He can only pull the box if he stands **exactly** 1 unit away from the box in the direction of one of two coordinate axes. He will pull the box to where he is standing before moving out of the way in the same direction by 1 unit.



For example, if the box is at the point $(1, 2)$ and Wabbit is standing at the point $(2, 2)$, he can pull the box right by 1 unit, with the box ending up at the point $(2, 2)$ and Wabbit ending at the point $(3, 2)$.

Also, Wabbit can move 1 unit to the right, left, up, or down without pulling the box. In this case, it is not necessary for him to be in exactly 1 unit away from the box. If he wants to pull the box again, he must return to a point next to the box. Also, Wabbit can't move to the point where the box is located.

Wabbit can start at any point. It takes 1 second to travel 1 unit right, left, up, or down, regardless of whether he pulls the box while moving.

Determine the minimum amount of time he needs to move the box from (x_1, y_1) to (x_2, y_2). Note that the point where Wabbit ends up at does not matter.

Input

Each test contains multiple test cases. The first line contains a single integer t ($1 \le t \le 1000$): the number of test cases. The description of the test cases follows.

Each of the next t lines contains four space-separated integers x_1, y_1, x_2, y_2 ($1 \le x_1, y_1, x_2, y_2 \le 10^9$), describing the next test case.

Output

For each test case, print a single integer: the minimum time in seconds Wabbit needs to bring the box from (x_1, y_1) to (x_2, y_2).

Examples

Input	Output
2 1 2 2 2 1 1 2 2	1 4

Note

In the first test case, the starting and the ending points of the box are $(1, 2)$ and $(2, 2)$ respectively. This is the same as the picture in the statement. Wabbit needs only 1 second to move as shown in the picture in the statement.

In the second test case, Wabbit can start at the point $(2, 1)$. He pulls the box to $(2, 1)$ while moving to $(3, 1)$. He then moves to $(3, 2)$ and then to $(2, 2)$ without pulling the box. Then, he pulls the box to $(2, 2)$ while moving to $(2, 3)$. It takes 4 seconds.

Problem G. University Classes

Time Limit 1000 ms
Mem Limit 262144 kB

There are n student groups at the university. During the study day, each group can take no more than 7 classes. Seven time slots numbered from 1 to 7 are allocated for the classes.

The schedule on Monday is known for each group, i. e. time slots when group will have classes are known.

Your task is to determine the minimum number of rooms needed to hold classes for all groups on Monday. Note that one room can hold at most one group class in a single time slot.

Input

The first line contains a single integer n $(1 \le n \le 1000)$ — the number of groups.

Each of the following n lines contains a sequence consisting of 7 zeroes and ones — the schedule of classes on Monday for a group. If the symbol in a position equals to 1 then the group has class in the corresponding time slot. In the other case, the group has no class in the corresponding time slot.

Output

Print minimum number of rooms needed to hold all groups classes on Monday.

Examples

Input	Output
2 0101010 1010101	1

Input	Output
3 0101011 0011001 0110111	3

Note

In the first example one room is enough. It will be occupied in each of the seven time slot by the first group or by the second group.

In the second example three rooms is enough, because in the seventh time slot all three groups have classes.

Problem H. Seating of Students

Time Limit 2000 ms
Mem Limit 262144 kB

Students went into a class to write a test and sat in some way. The teacher thought: "Probably they sat in this order to copy works of each other. I need to rearrange them in such a way that students that were neighbors are not neighbors in a new seating."

The class can be represented as a matrix with n rows and m columns with a student in each cell. Two students are neighbors if cells in which they sit have a common side.

Let's enumerate students from 1 to $n \cdot m$ in order of rows. So a student who initially sits in the cell in row i and column j has a number $(i - 1) \cdot m + j$. You have to find a matrix with n rows and m columns in which all numbers from 1 to $n \cdot m$ appear exactly once and adjacent numbers in the original matrix are not adjacent in it, or determine that there is no such matrix.

Input

The only line contains two integers n and m ($1 \le n, m \le 10^5$; $n \cdot m \le 10^5$) — the number of rows and the number of columns in the required matrix.

Output

If there is no such matrix, output "NO" (without quotes).

Otherwise in the first line output "YES" (without quotes), and in the next n lines output m integers which form the required matrix.

Examples

Input	Output
2 4	YES 5 4 7 2 3 6 1 8

Input	Output
2 1	NO

Note

In the first test case the matrix initially looks like this:

```
1 2 3 4
5 6 7 8
```

It's easy to see that there are no two students that are adjacent in both matrices.

In the second test case there are only two possible seatings and in both of them students with numbers 1 and 2 are neighbors.

Problem I. Case of Fugitive

Time Limit 3000 ms
Mem Limit 262144 kB

Andrewid the Android is a galaxy-famous detective. He is now chasing a criminal hiding on the planet Oxa-5, the planet almost fully covered with water.

The only dry land there is an archipelago of n narrow islands located in a row. For more comfort let's represent them as non-intersecting segments on a straight line: island i has coordinates $[l_i, r_i]$, besides, $r_i < l_{i+1}$ for $1 \le i \le n - 1$.

To reach the goal, Andrewid needs to place a bridge between each pair of **adjacent** islands. A bridge of length a can be placed between the i-th and the $(i + 1)$-th islads, if there are such coordinates of x and y, that $l_i \le x \le r_i$, $l_{i+1} \le y \le r_{i+1}$ and $y - x = a$.

The detective was supplied with m bridges, each bridge can be used at most once. Help him determine whether the bridges he got are enough to connect each pair of adjacent islands.

Input

The first line contains integers n ($2 \le n \le 2 \cdot 10^5$) and m ($1 \le m \le 2 \cdot 10^5$) — the number of islands and bridges.

Next n lines each contain two integers l_i and r_i ($1 \le l_i \le r_i \le 10^{18}$) — the coordinates of the island endpoints.

The last line contains m **integer** numbers $a_1, a_2, \ldots, a_m$ ($1 \le a_i \le 10^{18}$) — the lengths of the bridges that Andrewid got.

Output

If it is impossible to place a bridge between each pair of adjacent islands in the required manner, print on a single line `"No"` (without the quotes), otherwise print in the first line `"Yes"` (without the quotes), and in the second line print $n - 1$ numbers $b_1, b_2, \ldots, b_{n-1}$, which mean that between islands i and $i + 1$ there must be used a bridge number b_i.

If there are multiple correct answers, print any of them. Note that in this problem it is necessary to print "`Yes`" and "`No`" in correct case.

Examples

Input	Output
4 4 1 4 7 8 9 10 12 14 4 5 3 8	Yes 2 3 1

Input	Output
2 2 11 14 17 18 2 9	No

Input	Output
2 1 1 1 1000000000000000000 1000000000000000000 999999999999999999	Yes 1

Note

In the first sample test you can, for example, place the second bridge between points 3 and 8, place the third bridge between points 7 and 10 and place the first bridge between points 10 and 14.

In the second sample test the first bridge is too short and the second bridge is too long, so the solution doesn't exist.

Problem J. Fairy

Time Limit	1500 ms
Mem Limit	262144 kB
Input File	stdin
Output File	stdout

Once upon a time there lived a good fairy A. One day a fine young man B came to her and asked to predict his future. The fairy looked into her magic ball and said that soon the fine young man will meet the most beautiful princess ever and will marry her. Then she drew on a sheet of paper n points and joined some of them with segments, each of the segments starts in some point and ends in some other point. Having drawn that picture, she asked the young man to erase one of the segments from the sheet. Then she tries to colour each point red or blue so, that there is no segment having points of the same colour as its ends. If she manages to do so, the prediction will come true. B wants to meet the most beautiful princess, that's why he asks you to help him. Find all the segments that will help him to meet the princess.

Input

The first input line contains two integer numbers: n — amount of the drawn points and m — amount of the drawn segments ($1 \le n \le 10^4$, $0 \le m \le 10^4$). The following m lines contain the descriptions of the segments. Each description contains two different space-separated integer numbers v, u ($1 \le v \le n$, $1 \le u \le n$) — indexes of the points, joined by this segment. No segment is met in the description twice.

Output

In the first line output number k — amount of the segments in the answer. In the second line output k space-separated numbers — indexes of these segments in ascending order. Each index should be output only once. Segments are numbered from 1 in the input order.

Examples

Input	Output
4 4 1 2 1 3 2 4 3 4	4 1 2 3 4

Input	Output
4 5 1 2 2 3 3 4 4 1 1 3	1 5