Problem A. Love Triangle

Time limit 1000 ms
Mem limit 262144 kB

As you could know there are no male planes nor female planes. However, each plane on Earth likes some other plane. There are n planes on Earth, numbered from 1 to n, and the plane with number i likes the plane with number f_i, where $1 \le f_i \le n$ and $f_i \ne i$.

We call a love triangle a situation in which plane A likes plane B, plane B likes plane C and plane C likes plane A. Find out if there is any love triangle on Earth.

Input

The first line contains a single integer n $(2 \le n \le 5000)$ — the number of planes.

The second line contains n integers $f_1, f_2, \ldots, f_n$ $(1 \le f_i \le n, f_i \ne i)$, meaning that the i-th plane likes the f_i-th.

Output

Output «YES» if there is a love triangle consisting of planes on Earth. Otherwise, output «NO».

You can output any letter in lower case or in upper case.

Examples

Input	Output
5 2 4 5 1 3	YES

Input	Output
5 5 5 5 5 1	NO

Note

In first example plane 2 likes plane 4, plane 4 likes plane 1, plane 1 likes plane 2 and that is a love triangle.

In second example there are no love triangles.

Problem B. Party

Time limit	3000 ms
Mem limit	262144 kB
Input file	stdin
Output file	stdout

A company has n employees numbered from 1 to n. Each employee either has no immediate manager or exactly one immediate manager, who is another employee with a different number. An employee A is said to be the superior of another employee B if at least one of the following is true:

- Employee A is the immediate manager of employee B
- Employee B has an immediate manager employee C such that employee A is the superior of employee C.

The company will not have a managerial cycle. That is, there will not exist an employee who is the superior of his/her own immediate manager.

Today the company is going to arrange a party. This involves dividing all n employees into several groups: every employee must belong to exactly one group. Furthermore, within any single group, there must not be two employees A and B such that A is the superior of B.

What is the minimum number of groups that must be formed?

Input

The first line contains integer n ($1 \le n \le 2000$) — the number of employees.

The next n lines contain the integers p_i ($1 \le p_i \le n$ or $p_i = -1$). Every p_i denotes the immediate manager for the i-th employee. If p_i is -1, that means that the i-th employee does not have an immediate manager.

It is guaranteed, that no employee will be the immediate manager of him/herself ($p_i \ne i$). Also, there will be no managerial cycles.

Output

Print a single integer denoting the minimum number of groups that will be formed in the party.

Examples

Input	Output
5 -1 1 2 1 -1	3

Note

For the first example, three groups are sufficient, for example:

- Employee 1
- Employees 2 and 4
- Employees 3 and 5

Problem C. Li Hua and Maze

Time limit 1000 ms
Mem limit 262144 kB

There is a rectangular maze of size $n \times m$. Denote (r, c) as the cell on the r-th row from the top and the c-th column from the left. Two cells are *adjacent* if they share an edge. A *path* is a sequence of *adjacent* empty cells.

Each cell is initially empty. Li Hua can choose some cells (except (x_1, y_1) and (x_2, y_2)) and place an obstacle in each of them. He wants to know the minimum number of obstacles needed to be placed so that there isn't a *path* from (x_1, y_1) to (x_2, y_2).

Suppose you were Li Hua, please solve this problem.

Input

The first line contains the single integer t ($1 \le t \le 500$) — the number of test cases.

The first line of each test case contains two integers n, m ($4 \le n, m \le 10^9$) — the size of the maze.

The second line of each test case contains four integers x_1, y_1, x_2, y_2 ($1 \le x_1, x_2 \le n, 1 \le y_1, y_2 \le m$) — the coordinates of the start and the end.

It is guaranteed that $|x_1 - x_2| + |y_1 - y_2| \ge 2$.

Output

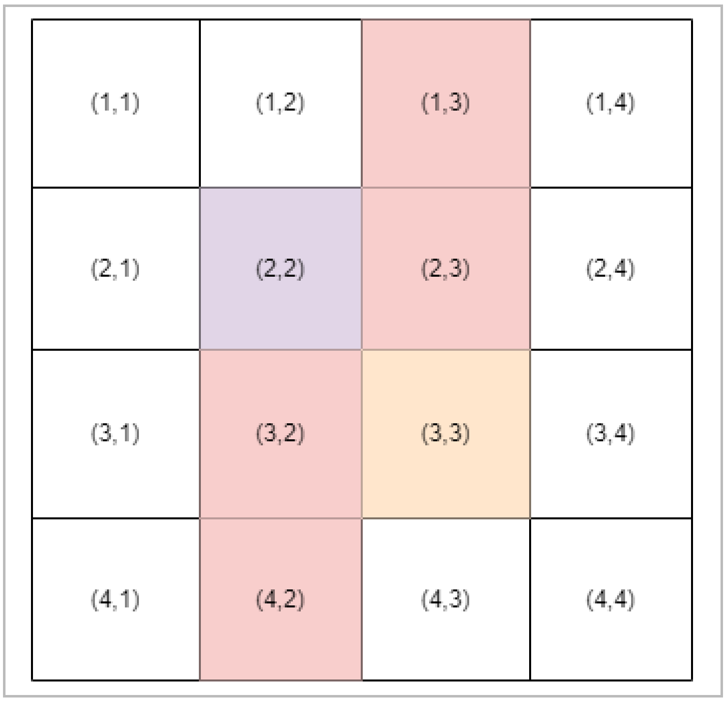
For each test case print the minimum number of obstacles you need to put on the field so that there is no *path* from (x_1, y_1) to (x_2, y_2).

Examples

Input	Output
3 4 4 2 2 3 3 6 7 1 1 2 3 9 9 5 1 3 6	4 2 3

Note

In test case 1, you can put obstacles on $(1,3)$, $(2,3)$, $(3,2)$, $(4,2)$. Then the path from $(2,2)$ to $(3,3)$ will not exist.



Problem D. Two Buttons

Time limit 2000 ms
Mem limit 262144 kB

Vasya has found a strange device. On the front panel of a device there are: a red button, a blue button and a display showing some positive integer. After clicking the red button, device multiplies the displayed number by two. After clicking the blue button, device subtracts one from the number on the display. If at some point the number stops being positive, the device breaks down. The display can show arbitrarily large numbers. Initially, the display shows number n.

Bob wants to get number m on the display. What minimum number of clicks he has to make in order to achieve this result?

Input

The first and the only line of the input contains two distinct integers n and m ($1 \le n, m \le 10^4$), separated by a space .

Output

Print a single number — the minimum number of times one needs to push the button required to get the number m out of number n.

Examples

Input	Output
`4 6`	`2`

Input	Output
`10 1`	`9`

Note

In the first example you need to push the blue button once, and then push the red button once.

In the second example, doubling the number is unnecessary, so we need to push the blue button nine times.

Problem E. Cthulhu

Time limit	2000 ms
Mem limit	262144 kB
Input file	stdin
Output file	stdout

...Once upon a time a man came to the sea. The sea was stormy and dark. The man started to call for the little mermaid to appear but alas, he only woke up Cthulhu...

Whereas on the other end of the world Pentagon is actively collecting information trying to predict the monster's behavior and preparing the secret super weapon. Due to high seismic activity and poor weather conditions the satellites haven't yet been able to make clear shots of the monster. The analysis of the first shot resulted in an undirected graph with n vertices and m edges. Now the world's best minds are about to determine whether this graph can be regarded as Cthulhu or not.

To add simplicity, let's suppose that Cthulhu looks from the space like some spherical body with tentacles attached to it. Formally, we shall regard as Cthulhu such an undirected graph that can be represented as a set of three or more rooted trees, whose roots are connected by a simple cycle.

It is guaranteed that the graph contains no multiple edges and self-loops.

Input

The first line contains two integers — the number of vertices n and the number of edges m of the graph ($1 \le n \le 100$, $0 \le m \le \frac{n \cdot (n-1)}{2}$).

Each of the following m lines contains a pair of integers x and y, that show that an edge exists between vertices x and y ($1 \le x, y \le n, x \ne y$). For each pair of vertices there will be at most one edge between them, no edge connects a vertex to itself.

Output

Print "`NO`", if the graph is not Cthulhu and "`FHTAGN!`" if it is.

Examples

Input	Output
6 6 6 3 6 4 5 1 2 5 1 4 5 4	FHTAGN!

Input	Output
6 5 5 6 4 6 3 1 5 1 1 2	NO

Note

Let us denote as a simple cycle a set of v vertices that can be numbered so that the edges will only exist between vertices number 1 and 2, 2 and 3, ..., v - 1 and v, v and 1.

A tree is a connected undirected graph consisting of n vertices and n - 1 edges ($n > 0$).

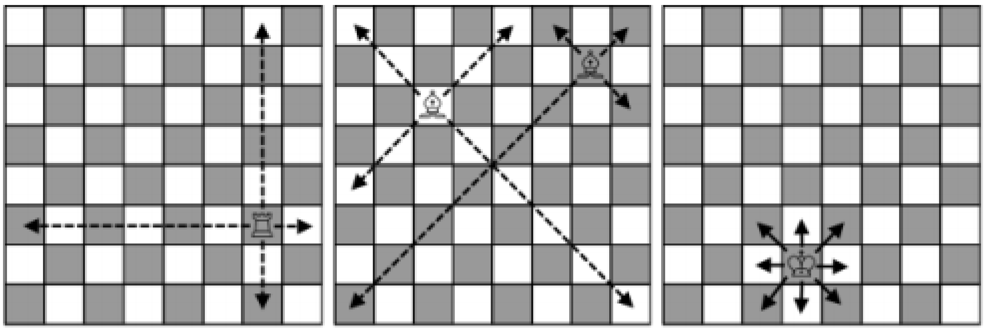
A rooted tree is a tree where one vertex is selected to be the root.

Problem F. Rook, Bishop and King

Time limit	1000 ms
Mem limit	262144 kB
Input file	stdin
Output file	stdout

Little Petya is learning to play chess. He has already learned how to move a king, a rook and a bishop. Let us remind you the rules of moving chess pieces. A chessboard is 64 square fields organized into an 8×8 table. A field is represented by a pair of integers (r, c) — the number of the row and the number of the column (in a classical game the columns are traditionally indexed by letters). Each chess piece takes up exactly one field. To make a move is to move a chess piece, the pieces move by the following rules:

- A rook moves any number of fields horizontally or vertically.
- A bishop moves any number of fields diagonally.
- A king moves one field in any direction — horizontally, vertically or diagonally.



The pieces move like that

Petya is thinking about the following problem: what minimum number of moves is needed for each of these pieces to move from field (r_1, c_1) to field (r_2, c_2)? At that, we assume that there are no more pieces besides this one on the board. Help him solve this problem.

Input

The input contains four integers r_1, c_1, r_2, c_2 $(1 \le r_1, c_1, r_2, c_2 \le 8)$ — the coordinates of the starting and the final field. The starting field doesn't coincide with the final one.

You can assume that the chessboard rows are numbered from top to bottom 1 through 8, and the columns are numbered from left to right 1 through 8.

Output

Print three space-separated integers: the minimum number of moves the rook, the bishop and the king (in this order) is needed to move from field (r_1, c_1) to field (r_2, c_2). If a piece cannot make such a move, print a 0 instead of the corresponding number.

Examples

Input	Output
`4 3 1 6`	`2 1 3`

Input	Output
`5 5 5 6`	`1 0 1`

Problem G. Minimum spanning tree for each edge

Time limit	2000 ms
Mem limit	262144 kB
Input file	stdin
Output file	stdout

Connected undirected weighted graph without self-loops and multiple edges is given. Graph contains n vertices and m edges.

For each edge (u, v) find the minimal possible weight of the spanning tree that contains the edge (u, v).

The weight of the spanning tree is the sum of weights of all edges included in spanning tree.

Input

First line contains two integers n and m ($1 \le n \le 2 \cdot 10^5$, $n - 1 \le m \le 2 \cdot 10^5$) — the number of vertices and edges in graph.

Each of the next m lines contains three integers u_i, v_i, w_i ($1 \le u_i, v_i \le n$, $u_i \ne v_i$, $1 \le w_i \le 10^9$) — the endpoints of the i-th edge and its weight.

Output

Print m lines. i-th line should contain the minimal possible weight of the spanning tree that contains i-th edge.

The edges are numbered from 1 to m in order of their appearing in input.

Examples

Input	Output
5 7 1 2 3 1 3 1 1 4 5 2 3 2 2 5 3 3 4 2 4 5 4	9 8 11 8 8 8 9