Problem A. Network Wars

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

Computer network of Byteland consists of n computers, connected by m cables. Every cable can transmit data in both directions. In Byteland a lot of important data is transmitted from the computer 1 (the King's computer) to the computer n (the General's computer).

Berland's commanders want to control all the data transmitted in Byteland. They want to buy some of the cables in such a way that every path from 1 to n through the cables contain at least one of the bought cables.

The cost of i-th cable is c_i. Berland' commanders want to buy cables in such a way that minimizes the average cost of the cable. In other words, if they buy k cables with total cost c, they want to minimize the value $\frac{c}{k}$.

Input

The first line of the input contains two integers n and m ($1 \le n \le 100$, $1 \le m \le 400$).

Next m lines contain descriptions of the cables — three integers a, b, c ($1 \le a, b \le n$, $1 \le c \le 10^7$), denoting two computers connected by the cable and its cost respectively.

It is guaranteed that no cable connects the computer with itself, there is at most one cable between every pair of computers. The network is connected, i.e. the data can be transmitted between each pair of computers using the cables.

Output

The first line should contain the number of cables Berland commanders need to buy.

The second line should contain the numbers of cables they need to buy in increasing order. The cables are numbered in the order they are described in the input.

If there are several optimal solutions, output any of them.

Examples

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

Problem B. Robots

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

One famous company has a rectangular warehouse with size $m \times n$ meters. It is splitted on rectangular cells 1×1 meter. There are several automatic robots with size of one cell, which can move inside of the warehouse. Robot can move to the adjacent cell (adjacent means that it has common edge). Some cells have obstacles inside it, so robot can't be there.

Wise CEO wants to use several robots, each robot should have it's own route. Routes shouldn't intersect, and each route should be a cycle (without self-intersections) of length greater than 2. Each cell without obstacle must be part of some route.

Input

First line contains 2 integers m and n ($1 \leqslant m, n \leqslant 50$).

Next m lines contain n integers each. Integer would be 1 if the cell contains obstacle, ans 0 if it's free.

Output

Print "`Yes`" if warehouse can be covered by such routes, and "`No`" otherwise.

Example

standard input	standard output
3 3 1 0 0 0 0 0 0 0 0	Yes
3 3 1 0 0 0 0 0 0 0 1	No

Problem C. Snow Removal

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

The capital of Berland contains n crossroads and m *unidirectional* roads. The i-th road connects crossroads x_i and y_i, is directed from x_i to y_i and has w_i tons of the snow.

The company "Snow so slow" has won the tender for snow removal. They are paid for each day of work, so they want to maximize the number of days they work.

On each day the company chooses one path from the crossroad A to the crossroad B using existing roads, which may pass through every crossroad or road more than one time. Then they remove one ton of snow from every road in the path. Note that if some road is used m times in the path, the amount of snow on it is decreased by m tons. Also note that in that case the road should contain at least m tons of snow before the removal (otherwise the company can be charged with corruption). For example, the company can not use the road which does not contain snow.

Some roads are valuable for government, they must be cleaned up after the work of the company. Other roads can contain snow even after the company finish its work.

You need to find a sequence of paths from A to B, which should be chosen by the company on each of the days, so that the number of days to work is maximized.

Input

The first line of the input contains four integers n, m, A, B ($2 \le n \le 100$, $0 \le m \le 5000$, $1 \le A, B \le n$, $A \ne B$) — the number of crossroads, the number of roads, the starting and the ending crossroads of all the paths.

Next m lines contain descriptions of the roads — four integers x_i, y_i, w, t_i ($1 \le x_i, y_i \le n$, $x_i \ne y_i$, $0 \le w_i \le 100$, $t_i \in \{0, 1\}$) — the start of i-th road, the end of i-th road, the amount of snow on the i-th road in tons and the type of the road (0 — usual road, 1 — valuable road).

It is guaranteed that for each ordered pair of cities there exists at most one road between them, i.e. $(x_i, y_i) \ne (x_j, y_j)$ for $i \ne j$.

Output

The first line should contain one integer p — the maximal number of days the company can work.

Next p lines contain the chosen paths. Each of them should be described as a sequence of crossroads' numbers, starting with A and ending with B.

If the solution does not exist, output 0.

Examples

standard input	standard output
4 7 1 4 1 2 3 1 2 1 100 0 2 4 1 0 1 3 1 0 3 4 4 0 2 3 2 1 1 4 2 0	6 1 3 4 1 4 1 4 1 2 4 1 2 3 4 1 2 3 4
3 3 1 2 1 3 2 0 3 2 3 0 1 2 1 0	3 1 3 2 1 3 2 1 2

Problem D. King and Roads

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	256 mebibytes

Most of Byteland's roads are not well-maintained. The king of Byteland, concerned by numerous requests of his subordinates, decided to renovate some of the roads. In Byteland there are n cities numbered 1 through n. Some pairs of cities are connected with *unidirectional* roads. The chief builder of Byteland has selected m roads which should be renovated and for each of those roads he estimated the renovation cost.

The king would like each citizen of Byteland to feel the difference in the quality of the road system. He assumed that citizens of a city will be satisfied if it is possible to both enter the city and leave the city using a renovated road. Your task is to select which roads to renovate to minimize the total renovation cost, while satisfying all citizens of Byteland.

Input

The first line of the input contains two integers n and m ($2 \leqslant n \leqslant 300$, $1 \leqslant m \leqslant n^2$) that represent the number of cities in Byteland and the number of unidirectional roads which can be renovated. Each of the next m lines of the input contains three integers x, y and k ($1 \leqslant x, y \leqslant n$, $0 \leqslant k \leqslant 10^5$) describing a road from the city x to the city y, which renovation cost is k dollars. Each ordered pair x, y appears in the input at most once. Observe that there might be a road starting and ending in the same city.

Output

The output should consist of a single line containing the minimum renovation cost, subject to the constraints mentioned in the task description, or a single word "`NIE`" (Polish for *no*), if it is impossible to design a renovation plan fulfilling king's requirements.

Example

standard input	standard output
4 6 1 2 1 2 1 2 1 3 3 3 1 4 3 2 5 4 4 6	16
4 4 1 2 5 2 3 4 3 1 8 2 4 7	NIE

Problem E. Spy Union

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	256 mebibytes

The spy network of the WSA has grown too big, downsizing is unavoidable. Unfortunately the Trade Union of Spies is large and strong, and they have spies everywhere. After many months of fruitless fighting, the management of the WSA met the representatives of the union, and together they came up with two directed trees.

The first tree represents the hierarchy of the WSA. Each employee is a node, and each node is the head of a department (the node together with all its descendants). Each node is tagged with an integer that specifies how many employees have to be in that department to keep the organization operational. The second tree is the hierarchy of the union — a different hierarchy with different integers, but the meaning of the tree is the same. All employees are present in both trees.

Your task is to determine which employees to dismiss to get the smallest possible organization while still keeping all WSA and union departments operational.

Note that department heads may be fired — in which case a subordinate will be the new head; but some departments may have a tag of 0, meaning that the department is superfluous and may be eliminated altogether.

Input

The first line of the input contains an integer N ($1 \le N \le 2 \cdot 10^4$), the number of employees. Their Employee IDs (EIDs) are integers between 0 and $N-1$. The next N lines refer to employees, the i-th line to the employee with EID i, in B_w B_u R_w R_u format. B_x, where $x = w$ or $x = u$, is the EID of this employee's boss in the w (WSA) or u (union) hierarchies ($0 \le B_x < N$, the boss EID of the top manager is their own number). R_x is the number of required people in this employee's respective subtree ($0 \le R_x \le N$).

Output

The first line should contain an integer K, the maximum number of employees who can be fired.

The second line should contain K integers, the EIDs of the dismissed employees.

Example

standard input	standard output
5 1 0 1 2 2 0 1 2 2 1 2 0 2 1 0 1 1 3 0 0	2 4 2