

BOI 97 - Street Network

The street network of a city is composed of streets and nodes. In a node, two or more streets can meet. All streets are one-way streets. Note also that, two nodes can be connected directly by more than one street, and one node can have a street that loops back to itself. Write a computer program in order to address the following issues: 1. Is it possible to start from at least one node A and visit ALL streets exactly once ? 2. How many nodes can serve as starting points in order to satisfy the property of the previous case ? 3. For each node X, how many paths of length S exist starting from X and ending to X, where any street or node can be visited more than once ?

Input

In the first line in the input is a positive integer number N ($N \leq 50$), denoting the number of nodes in the city street network. The second line contains a positive integer number S ($S \leq 3$) denoting the path length. The next N lines contain the network description in matrix form. More precisely, the element in row I and column J is the number of streets from node I to node J.

Output

The first line contains the string "YES" if you can start from a node, travel through all streets exactly once, and arrive either at the starting point, or at another node. Otherwise, the string "NO" should appear in the output. If the answer is "YES", the next line of the output file should contain a positive integer number denoting how many nodes can serve as starting points. Finally, the last line of the output file should contain N positive integers (separated by a space) that show for each node how many different paths with length S exist such that each path leads from the node back to itself. These numbers should be sorted in increasing order.

Example

Input 1:

```
3
2
1 1 0
1 1 1
0 1 1
```

Output:

```
YES
3
2 2 3
```

Input 2:

```
3
2
1 1 0
1 1 2
0 0 1
```

Output:

```
NO
```

