
Factory

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

It is suspected, that the military factory on the outskirts of the town is connected with a series of recent crimes. Public Safety Bureau sent inspector Akane to find out if that was the case.

Akane has a rectangular map of the factory, which is a table of size $m \times n$. Each cell of the map is either empty, or contains one of the factory's workshops. Of course, all workshops are cell-connected. That is, it's possible to get from any workshop to any other workshop through several intermediate workshops, going only between workshops sharing a side. It's also guaranteed, that the factory doesn't enclose any area that doesn't belong to the factory. That is, it's possible to get from any non-workshop cell to the border of the map, going only through non-workshop cells sharing a side.

Akane believes, that in case there are some clues to find, it's more likely that they are located in the corner of some workshop, rather than in it's center. For the above-mentioned map consider all grid nodes that are at the corner of at least one workshop. Akane calls such nodes important. It's easy to see, that the number of important nodes is at most $(m + 1) \cdot (n + 1)$. Akane also found out, that there are corridors on the border of each workshop cell connecting adjacent grid nodes at the corners of this cell.

Akane wants to traverse all the important nodes and to do it in a most efficient way. More precisely, Akane wants to make a route going through the nodes of the grid, such that:

- the route starts and ends at the same node;
- between every two nodes Akane goes through a single corridor;
- each important node is visited at least once;
- route goes through important nodes only;
- Akane goes through every corridor at most once.

Please help Akane find any valid route or determine, that it's impossible. Please note, that it's not required to minimize the length of the route.

Input

The first line contains integers m and n ($1 \leq m, n \leq 20$) — the size of the map with the factory.

The following m lines contain n characters each and describe the factory's map. The character "*" denotes the workshop, while "." denotes an empty cell.

Output

In case it's impossible to traverse the factory, print "No".

Otherwise print "Yes" and then the number of corridors Akane will go through.

Then print the route itself, one grid node per line. Let's number all horizontal grid lines from 0 to m upside down and number all vertical grid lines from 0 to n from left to right. Each grid node is described with two integers r_i and c_i ($0 \leq r_i \leq m, 0 \leq c_i \leq n$) — the row number and the column number.

Examples

| standard input | standard output |
|----------------------------|---|
| <pre>3 3 *** *** .**</pre> | <pre>Yes 16 0 0 0 1 1 1 1 2 1 3 0 3 0 2 1 2 2 2 2 3 3 3 3 2 3 1 2 1 2 0 1 0</pre> |
| <pre>1 4 ****</pre> | <pre>Yes 10 0 0 0 1 0 2 0 3 0 4 1 4 1 3 1 2 1 1 1 0</pre> |
| <pre>2 2 ** **</pre> | <pre>No</pre> |

Note

Akane's route in the first example is as follows:

