## The Goal

The game is played on a rectangular grid with a given size. Some cells contain power nodes. The rest of the cells are empty.  
  
The goal is to find, when they exist, the horizontal and vertical neighbors of each node.

## Rules

To do this, you must find each (x1,y1) coordinates containing a node, and display the (x2,y2) coordinates of the next node to the right, and the (x3,y3) coordinates of the next node to the bottom within the grid.  
  
If a neighbor does not exist, you must output the coordinates instead of (x2,y2) and/or (x3,y3).  
  
You lose if:

* You give an incorrect neighbor for a node.
* You give the neighbors for an empty cell.
* You compute the same node twice.
* You forget to compute the neighbors of a node.

Victory Conditions

You win when all nodes have been correctly displayed.

## Example



In this example, there are three nodes in a 2 by 2 grid. The cell at (1,1) is empty.

00

0.

The node at (0,0) has 2 neighbors.



The node at (1,0) has no neighbors.



The node at (0,1) has no neighbors.

## Note

Don’t forget to run the tests by launching them from the “Test cases” window.  
  
Warning: the tests provided are similar to the validation tests used to compute the final score but remain different. This is a "hardcoding" prevention mechanism. Harcoded solutions will not get any points.

Regarding the viewer, note that:

* A debug mode is available from the settings panel (the dented wheel)
* You can zoom/unzoom with the mouse wheel and move using drag'n drop (useful for large grids)

## Game Input

The program must first read the initialization data from standard input. Then, provide to the standard output one line per instruction.

Initialization input

Line 1: one integer width for the number of cells along the x axis.

Line 2: one integer height for the number of cells along the y axis.

Next height lines: A string  line  containing  width  characters. A dot represents an empty cell. A zero represents a cell containing a node.

Output for one game turn

One line per node. Six integers on each line:   x1  y1  x2  y2  x3  y3  
  
Where:

* (x1,y1) the coordinates of a node
* (x2,y2) the coordinates of the closest neighbor on the right of the node
* (x3,y3) the coordinates of the closest bottom neighbor

If there is no neighbor, the coordinates should be .

Constraints

0 < width ≤ 30  
0 < height ≤ 30  
0 ≤ x1 < width  
0 ≤ y1 < height  
-1 ≤ x2, x3 < width  
-1 ≤ y2, y3 < height  
Alloted response time to first output line ≤ 1s.  
Response time between two output lines ≤ 100ms