B. Infinite Maze

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

We've got a rectangular $n \times m$-cell maze. Each cell is either passable, or is a wall (impassable). A little boy found the maze and cyclically tiled a plane with it so that the plane became an infinite maze. Now on this plane cell (x, y) is a wall if and only if cell is a wall.

In this problem is a remainder of dividing number a by number b.

The little boy stood at some cell on the plane and he wondered whether he can walk infinitely far away from his starting position. From cell (x, y) he can go to one of the following cells: $(x, y - 1)$, $(x, y + 1)$, $(x - 1, y)$ and $(x + 1, y)$, provided that the cell he goes to is not a wall.

Input

The first line contains two space-separated integers n and m ($1 \leq n, m \leq 1500$) — the height and the width of the maze that the boy used to cyclically tile the plane.

Each of the next n lines contains m characters — the description of the labyrinth. Each character is either a "#", that marks a wall, a ".", that marks a passable cell, or an "S", that marks the little boy's starting point.

The starting point is a passable cell. It is guaranteed that character "S" occurs exactly once in the input.

Output

Print "Yes" (without the quotes), if the little boy can walk infinitely far from the starting point. Otherwise, print "No" (without the quotes).

Examples

input
5 4 ##.# ##S# #..# #.## #..#

output
Yes

input
5 4 ##.# ##S# #..# ..#. #.##

output
No

Note

In the first sample the little boy can go up for infinitely long as there is a "clear path" that goes vertically. He just needs to repeat the following steps infinitely: up, up, left, up, up, right, up.

In the second sample the vertical path is blocked. The path to the left doesn't work, too — the next "copy" of the maze traps the boy.