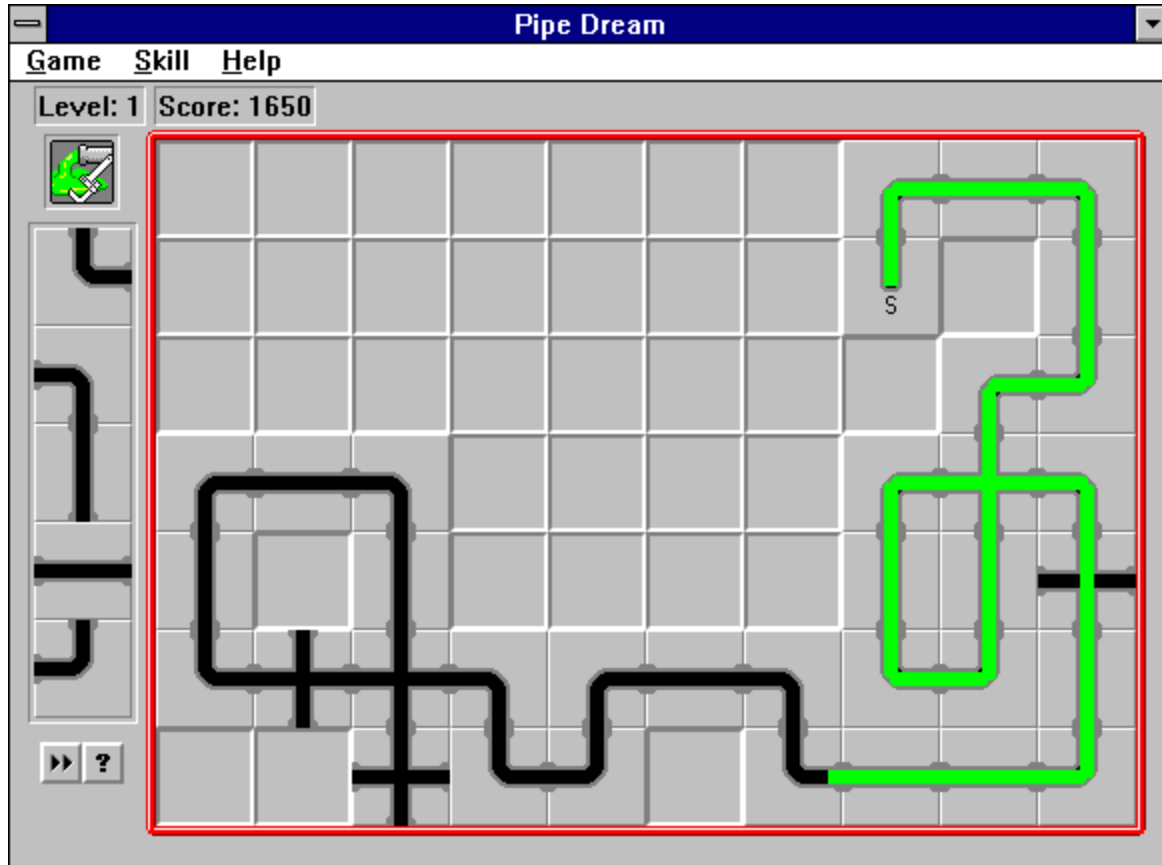


Pipe Dream

Time limit: 1.0s **Memory limit:** 16M




As a kid, you might have played the game *Pipe Dream*.
Here's a picture (yes, from Windows 3.1!):



The goal is basically to get the fluid from one place to another.
In the original game, the pieces came to you randomly and you had to create the longest path.

However, interesting games for humans tend to be hard for computers. Thus, your task is easier: use pipes to just guide the flow from one point to another. You will only have 4 types of pipes, and you can use as many of them wherever you wish.

You need to use the four "angle" pipes - use the characters  to represent them. To output these, use the following UTF-8 sequences:

| Character | UTF-8 sequence (hex) |
|---|----------------------|
|  | E2 95 94 |
|  | E2 95 97 |
|  | E2 95 9A |

| Character | UTF-8 sequence (hex) |
|-----------|----------------------|
| ⌋ | E2 95 9D |

A grid with obstacles and the source/destination blocks will be given to you. Output any structure of pipes that gets the fluid to its destination.

Input Specification

R (rows) ≤ 100 , C (columns) ≤ 100 , the size of the grid.

A grid of R lines with C characters each.

A will represent a wall, the source, the destination, and an empty spot.

There will only be one source, and one destination.

Output Specification

Arrange some pipes (if necessary) so that fluid can get from to .

If it's impossible, just output .

Sample Input 1

```

9 12
#####
#       #
#     S #
#       #
#       #
#       #
#       #
#       #
#       #
###D#####

```

Sample Output 1

```
#####
#       █ #
#       S█#
#       █#
#       █#
#  █     █ #
# █ █ █ █ █ #
# █ █ █ █ █ #
#####D#####
```

A similar setup to the picture (not quite as interesting, since you can't have straight pipes).
Your output need not be so complex, though.

Sample Input 2

```
5 10
#####
#S   # ##
#   #####
###   #
#####D#####
```

Sample Output 2

```
#####
#S█  # ##
# █ #####
###█  #
#####D#####
```

Of course, walls could be anywhere.