Shortest Path in Grid

Time limit: 1 sec

We fall down into a maze that can be described by a grid of R rows and C columns. Each row and column are numbered from 1 (top row) to R (bottom row) and from 1 (left-most column) to C (right-most column). Hence, a cell located at row r and column c can be described by a coordinate (r, c).

The cell that we currently fall into is (1, 1). The ladder that allow us to climb out of the maze is at cell (R,C). We would like to take a shortest path to that cell. Each cell in the maze is either an empty cell or an obstacle cell. We can walk from an empty cell to any adjacent empty cell that share a side with our cell. However, we cannot walk into an obstacle cell.

Your task is to find a shortest path that leads us from cell (1, 1) to cell (R, C). Assuming that cell (1, 1) is always an empty cell. The length of the path is the number of cells that it passes. It is also possible that there is no way to go to cell (R, C). We should be able to detect this case as well.

Input

- The first line of input contains two integers that give the size of the grid **R** and **C**. (1 <= R, C <= 100).
- The next **R** lines describe the maze. Each line describes a row, starting from row 0 to row **R**. The format of each line is as follow.
    - Each line contains a string of length **C**. Each character in the string represents a cell in the corresponding row, starting from the column 0 to column **C**. A character '**#**' indicates that the corresponding cell is an obstacle cell while a character '**.**' indicates that the cell is an empty cell.

Output

The output must contain exactly one line giving the length of the shortest path that leads us to the cell (R, C). If there is no such path, the line must be '-1'.

Example

Input	Output
5 5 .#... .#... .#... .#.#. ...#.	12
5 5 ....## ...#. ..##. .###.	-1