Pogo

This question is graded for 0.5%!

Statement

Fluffy is navigating a maze. The maze can be denoted as a R by C grid of cells, where each cell is either a blank space or an obstacle. A blank cell is denoted by `.`, while a cell with an obstacle is denoted by `#`. In one minute, Fluffy can move to adjacent cells that share a border, but not to cells with obstacles.

To help himself get to the destination quicker, Fluffy has brought a pogo stick with him. If an adjacent cell is occupied by an obstacle, Fluffy can use his pogo stick to hop over the obstacle to the next adjacent cell in the same direction, if that cell does not contain an obstacle. This action can also be performed in one minute.

Fluffy starts from the cell denoted by `S`, and would like to get to the cell denoted by `E`. Help him find the minimum amount of time required to reach his destination.

Constraints

- $1 \leq R, C \leq 2^{10}$
- There will be exactly one copy of `S` and `E` each in the grid.

Input

The first line of input will contain two integers R and C.

The next R lines of input will each contain C characters. Each character is either `S`, `E`, `.` or `#`.

Output

Print a single integer, the minimum time required for Fluffy to move from `S` to `E`.

Examples

Sample Input	Expected Output

`4 5`
`S#.#.`
`####.`
`#E###`
`#..#.`
	`7`

Notes

1. A skeleton file has been given to help you. You should not create a new file or rename the file provided. You should develop your program using this skeleton file.

2. You are free to define your own helper methods and classes (or remove existing ones) if it is suitable but you must put all the new classes, if any, in the same skeleton file provided.

Skeleton File

You are given the skeleton file `Pogo.java`. You should see the following contents when you open the file:

```
/**
 * Name         :
 * Matric. No   :
*/

import java.util.*;

public class Pogo {
    private void run() {
    }

    public static void main(String args[]) {
        Pogo runner = new Pogo();
        runner.run();
    }
}
```