



Overfencing

Kolstad and Schrijvers

Farmer John went crazy and created a huge maze of fences out in a field. Happily, he left out two fence segments on the edges, and thus created two "exits" for the maze. Even more happily, the maze he created by this overfencing experience is a 'perfect' maze: you can find a way out of the maze from any point inside it.

Given W ($1 \leq W \leq 38$), the width of the maze; H ($1 \leq H \leq 100$), the height of the maze; $2*H+1$ lines with width $2*W+1$ characters that represent the maze in a format like that shown later - then calculate the number of steps required to exit the maze from the 'worst' point in the maze (the point that is 'farther' from either exit even when walking optimally to the closest exit). Of course, cows walk only parallel or perpendicular to the x-y axes; they do not walk on a diagonal. Each move to a new square counts as a single unit of distance (including the move "out" of the maze).

Here's what one particular $W=5$, $H=3$ maze looks like:

```

+--+--+--+
|      |
+--+--+--+
|      |
+--+--+--+
|      |
+--+--+--+

```

Fenceposts appear only in odd numbered rows and odd numbered columns (as in the example). The format should be obvious and self explanatory. Each maze has exactly two blank walls on the outside for exiting.

PROGRAM NAME: maze1

INPUT FORMAT

| | |
|---------------------------|--|
| Line 1: | W and H , space separated |
| Lines 2 through $2*H+2$: | $2*W+1$ characters that represent the maze |

SAMPLE INPUT (file maze1.in)

```

5 3
+--+--+--+
|      |
+--+--+--+
|      |
+--+--+--+
|      |
+--+--+--+

```

OUTPUT FORMAT

A single integer on a single output line. The integer specifies the minimal number of steps that guarantee a cow can exit the maze from any possible point inside the maze.

SAMPLE OUTPUT (file maze1.out)

9

The lower left-hand corner is **nine** steps from the closest exit.

Submit a solution:

Choose File

No file chosen

Send it in!

[Training Gateway](#) | [Comment or Question](#)