Problem G : Mona and the treasure

Statement:

Mona has heard about a famous treasure that is hidden in the castle so naturally she wanted to claim it as her own. She went to the castle but she found that in order to get to the treasure, she first has to go through a maze in a room. However it turned out that the shape of this maze is quite easy so she wants to calculate the minimum distance she has to travel to get to the end.

The maze is an **n*m** grid consisting of walls (rows) she has to traverse one by one. Each wall has a single door she can traverse through.

The shape of the maze is given as follows :

Each line contains the description of a wall, the doors are marked with '**.**' and the rest of the cells of the wall are marked with '#'.

- Mona always starts at the 1st cell of the 1st wall.
- The 1st cell of the 1st wall is always a door.
- Mona has to get to the end wich is the last cell of the last wall.
- The last cell of the last wall is always a door.
  ```
  <pre>
  .######
  ######.
  #####.#
  #####.#
  #.#####
  .######
  ######.
  </pre>
  ```

Your task is to find out the minimum number of steps she has to take in order to get to the end. (Suppose that moving horizontally by a single cell counts as one move and moving vertically doesn't count as a move)

Input :

The first line contains a single integer **T** (1≤T≤100) — the number of test cases. Then the test cases follow. Each test case consists of one line.

The first line contains two integers **n**, **m** (1≤n, m≤100) — where **n** is the number of walls and **m** is the width of these walls.

Each of the next n lines contains m characters. The **i**th line describes the **i**th wall. If a character on a line equals ".", then the corresponding cell is a door and if the character equals "#", then the cell is part of the wall.

Output :

For each test case, output the minimum number of steps she has to make in order to get to the end. (Suppose that moving horizontally by a single cell counts as one move and moving vertically doesn't count as a move)

Example:

Input :

```
2
3 4
.###
###.
###.
7 7
.######
######.
#####.#
#####.#
#.#####
.######
######.
```

Output :

```
3
18
```