Problem B : Luffy is a Foodie

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Monkey D. Luffy, the main character of “One Piece,” is known for his love of food, particularly meat.

You will be given a string consisting of the characters ‘**.**’, ‘**#**’ and ‘**|**’:
‘**.**’ is an empty spot, ‘**#**’ is a piece of meat and ‘**|**’ is a wall. This string represents a **1D** grid.

Luffy needs to choose a starting point and a walking direction(left or right) such that he maximizes the meat eaten.

So he asked for your help.

Your goal is to put Luffy in one of the empty spots in order to maximize the meat eaten such that:

- Luffy can move in only 1 direction in the grid(left or right), in other words he can’t change his direction after choosing it.
- Luffy will stop if he finds a wall or the edge of the grid.

It is guaranteed that the grid contains at least one empty spot.

Input :

The first line of input contains a single integer **T** ($1 \le T \le 100$) — the number of test cases. Then the test cases follow. Each test case consists of two lines.

The first line of each test case contains an integer **n** ($1 \le n \le 10^6$) — the length of the string

The second line of each test case contains a string **s** of length **n**, consisting of characters ‘**#**’, ‘**.**’, ‘**|**’.

It is guaranteed that the sum of **n** over all test cases doesn’t exceed 10^6.

Output :

For each test case, output the maximum quantity of meat Luffy can get.

Example:

Input :

```
2
8
##.#|#.#
12
|.#|#.#.#..#
```

Output :

```
2
3
```

Note:

In the **first** testcase, the first empty spot is picked (position with index 3), the direction that maximizes the answer is left. In the **second** testcase, there are 2 possible empty spots, the 4th empty spot (position with index 10) and the 5th empty spot (position with index 11), the direction that maximizes the answer is left.