

# Teleporters and Gems

Duck is playing "Teleporters and Gems". On a  $1 \times N$  map, there may be multiple teleporters (or no) and at least one gem. At the beginning, Duck is standing at the most left position. He has to collect all the gems with the help of teleporters. Below are the rules:

1. Moving 1 unit is counted as 1 move
2. When Duck reaches a teleporter, he can instantly reach any of the next 3 teleporters (eg. 1 to 2, 3 or 4 teleporter, but not 1 to 5, 6, 7 teleporter, and previous not allowed) by 3 moves; Or he can ignore that teleporter and keep moving to next cell which is counted as 1 move

Let '@' = Teleporter, '.' = empty cell, there are 4 teleporters: @.@..@@, if Duck is at position 0 (the 1-st teleporter), then he can instantly reach 2-nd, 3-rd or 4-th teleporter by 3 moves. But ignoring the 1-st teleporter makes him reach 2-nd teleporter by 2 moves.

3. Duck has to collect all gems
4. Once he collects the last gem, the game ends and no more move needed
5. Duck can only move right

Can you help him find out the minimum number of moves in order to collect all gems?

## Input

The first line is the number of test cases **T**. ( $1 \leq T \leq 50$ )

For each test case, there is a string **S** consisting of '.' ( $\geq 0$ ), '@' ( $\geq 0$ ) and '\*' ( $\geq 1$ ), representing empty cell, teleporter and gem. ( $1 \leq |S| \leq 10^4$ )

## Output

Print one integer  $x$  where  $x$  is the minimum number of moves in order to collect all gems starting from the most left cell.

## Example

**Input:**

```
3
.@@...@.@...@@.*@.@.*...@..
....@...@@.**.@...@@@*@.
.*....*@@@@*..@*..@
```

**Output:**

```
14
16
16
```

## Explanation

Bracket means cells that Duck will pass through and minimum number of moves to that cell

Case 1: (.@@)...@.@...(@@..\* @.@.\*)...@.. = 0 1 2 -> 5 6 7 8 9 10 11 12 13 14

Case 2: (...@)...@(@.\*\*.@)...@@(@\*)@. = 0 1 2 3 4 -> 7 8 9 10 11 12 -> 15 16

Case 3: (. \* ... \* @)@@(@\*..@@\*)..@ = 0 1 2 3 4 5 6 7 -> 10 11 12 13 14 15 16