

En Route Salute

=====

Commander Lambda loves efficiency and hates anything that wastes time. She's a busy lamb, after all! She generously rewards henchmen who identify sources of inefficiency and come up with ways to remove them. You've spotted one such source, and you think solving it will help you build the reputation you need to get promoted.

Every time the Commander's employees pass each other in the hall, each of them must stop and salute each other - one at a time - before resuming their path. A salute is five seconds long, so each exchange of salutes takes a full ten seconds (Commander Lambda's salute is a bit, er, involved). You think that by removing the salute requirement, you could save several collective hours of employee time per day. But first, you need to show her how bad the problem really is.

Write a program that counts how many salutes are exchanged during a typical walk along a hallway. The hall is represented by a string. For example:

"--->-><-><-->--"

Each hallway string will contain three different types of characters: '>', an employee walking to the right; '<', an employee walking to the left; and '-', an empty space. Every employee walks at the same speed either to right or to the left, according to their direction. Whenever two employees cross, each of them salutes the other. They then continue walking until they reach the end, finally leaving the hallway. In the above example, they salute 10 times.

Write a function `solution(s)` which takes a string representing employees walking along a hallway and returns the number of times the employees will salute. `s` will contain at least 1 and at most 100 characters, each one of -, >, or <.

Languages

=====

To provide a Python solution, edit `solution.py`

To provide a Java solution, edit `Solution.java`

Test cases

=====

Your code should pass the following test cases.

Note that it may also be run against hidden test cases not shown here.

-- Python cases --

Input:

```
solution.solution(">----<")
```

Output:

2

Input:

```
solution.solution("<<>><")
```

Output:

4

-- Java cases --

Input:

```
Solution.solution("<<>><")
```

Output:

4

Input:

```
Solution.solution(">----<")
```

Output:

2

Use `verify [file]` to test your solution and see how it does. When you are finished editing your code, use `submit [file]` to submit your answer. If your solution passes the test cases, it will be removed from your home folder.