

Coding Competitions Farewell Rounds - Round A

Untie

Problem

A group of people are sitting in a circle, playing a special version of rock, paper, scissors. In this game, each person chooses [rock, paper, or scissors](#) in secret and then everyone reveals their choice to everyone else. Each person then compares their selection to their two neighbors, and can win, lose, or tie against each of them independently. The only way to tie is when both people make the same choice.

You want to make it so that no game is a tie. For each player, you can let them keep their choice, or you can ask them to change to any of the other two options (you choose to which one). What is the minimum number of people you need to request a change from to ensure that there are no ties between neighbors after those changes are made?

Input

The first line of the input gives the number of test cases, T . T lines follow. Each line represents a test case and contains a string C . The i -th character of C represents the original choice of the i -th person in clockwise order using an uppercase R to mean rock, an uppercase P to mean paper, and an uppercase S to mean scissors.

Output

For each test case, output one line containing `Case # x : y` , where x is the test case number (starting from 1) and y is the minimum number of changes that are required such that no two neighbors end up with the same choice.

Limits

Time limit: 10 seconds.

Memory limit: 2 GB.

$1 \leq T \leq 100$.

Each character of C is either an uppercase R , an uppercase P , or an uppercase S .

Test Set 1 (Visible Verdict)

$3 \leq \text{the length of } C \leq 10$.

Test Set 2 (Visible Verdict)

$3 \leq \text{the length of } C \leq 1000$.

Sample

Sample Input

```
3
PRSSP
```

Sample Output

```
Case #1: 2
Case #2: 4
```

RRRRRRR
RSPRPSPRS

Case #3: 0

In Sample Case #1, there is a pair of neighbors that both chose paper (the first and last character of the input) and another pair that both chose scissors. Therefore, we need at least two changes. One way of doing it with two changes is to change the leftmost paper to scissors and the rightmost scissors to rock, to obtain `SRSRP`.

In Sample Case #2, all 7 participants chose rock. If we change at most 3 selections, there will be at least 4 remaining rocks, and at least two of them will be neighbors. Therefore, the minimum number of changes is at least 4. One way to achieve exactly 4 is to get `PRSRPRS`.

In Sample Case #3, no pair of neighbors tied, so no changes are needed.