Counting Valleys ☆

Your Counting Valleys submission got 15.00 points. Share Tweet

Try the next challenge

Problem Submissions Leaderboard Editorial

Gary is an avid hiker. He tracks his hikes meticulously, paying close attention to small details like topography. During his last hike he took exactly n steps. For every step he took, he noted if it was an *uphill*, U, or a *downhill*, D step. Gary's hikes start and end at sea level and each step up or down represents a 1 unit change in altitude. We define the following terms:

- A *mountain* is a sequence of consecutive steps *above* sea level, starting with a step *up* from sea level and ending with a step *down* to sea level.
- A *valley* is a sequence of consecutive steps *below* sea level, starting with a step *down* from sea level and ending with a step *up* to sea level.

Given Gary's sequence of *up* and *down* steps during his last hike, find and print the number of *valleys* he walked through.

For example, if Gary's path is $s = [DDUUUUDD]$, he first enters a valley 2 units deep. Then he climbs out an up onto a mountain 2 units high. Finally, he returns to sea level and ends his hike.

Function Description

Complete the *countingValleys* function in the editor below. It must return an integer that denotes the number of valleys Gary traversed.

countingValleys has the following parameter(s):

- *n*: the number of steps Gary takes
- *s*: a string describing his path

Input Format

The first line contains an integer n, the number of steps in Gary's hike.
The second line contains a single string s, of n characters that describe his path.

Constraints

- $2 \le n \le 10^6$
- $s[i] \in \{UD\}$

Output Format

Print a single integer that denotes the number of valleys Gary walked through during his hike.

Sample Input

```
8
UDDDUDUU
```

Sample Output

```
1
```

Explanation

If we represent _ as sea level, a step up as / , and a step down as \ , Gary's hike can be drawn as:

```
_/\      _
   \    /
    \/\/
```

He enters and leaves one valley.

Java 7

```java
import java.io.*;
import java.math.*;
import java.security.*;
import java.text.*;
import java.util.*;
import java.util.concurrent.*;
import java.util.regex.*;

public class Solution {

    // Complete the countingValleys function below.
    static int countingValleys(int n, String s) {

        int valleys=0;
        int level=0;
        for(char c: s.toCharArray())
        {
            if(c=='U')
                ++level;
            if(c=='D')
                --level;
            if(level==0 && c=='U')
                valleys++;

        }
    return valleys;

    }

    private static final Scanner scanner = new Scanner(System.in);

    public static void main(String[] args) throws IOException {
        BufferedWriter bufferedWriter = new BufferedWriter(new FileWriter(System.getenv("OUTPUT_PATH")));

        int n = scanner.nextInt();
        scanner.skip("(\r\n|[\n\r\u2028\u2029\u0085])?");

```

Line: 22 Col: 34

Run Code Submit Co

↑ Upload Code as File ☐ Test against custom input

Upload Code as File

You have earned 15.00 points!

These points will also count towards your progress in the Problem Solving Badge.

83% 25/30

Problem Solving

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

- Test case 0 ✓
- Test case 1 ✓
- Test case 2 ✓
- Test case 3 ✓
- Test case 4 ✓
- Test case 5 ✓
- Test case 6 ✓

Compiler Message

```
Success
```

Input (stdin) Download

```
8
UDDDUDUU
```

Expected Output Download

```
1
```