An avid hiker keeps meticulous records of their hikes. During the last hike that took exactly ***steps***  steps, for every step it was noted if it was an *uphill*, U , or a *downhill*, D step. Hikes always 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 the sequence of *up* and *down* steps during a hike, find and print the number of *valleys* walked through.

**Example**

The hiker first enters a valley  units deep. Then they climb out and up onto a mountain  units high. Finally, the hiker returns to sea level and ends the hike.

**Function Description**

Complete the *countingValleys* function in the editor below.

countingValleys has the following parameter(s):

* *int steps*: the number of steps on the hike
* *string path*: a string describing the path

**Returns**

* *int:* the number of valleys traversed

**Input Format**

The first line contains an integer ***steps*** , the number of steps in the hike.  
The second line contains a single string ***path*** , of  ***steps*** characters that describe the path.

**Constraints**

* 2 <= steps <= 106
* Path[i] E {UD}

**Sample Input**

8

UDDDUDUU

**Sample Output**

1

**Explanation**

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

\_/\ \_

\ /

\/\/

The hiker enters and leaves one valley.

Um caminhante ávido mantém registros meticulosos de suas caminhadas. Durante a sua última caminhada que levou exatamente ***steps*** passos, para cada etapa ele anotou se era subida, ***U***, ou descida, ***D***, passos. Toda caminhada começa no nível do mar e cada passo para cima ou para baixo representa 1 unidade na altitude. Definimos os seguintes termos:

* Uma montanha é uma sequência consecutiva de degraus acima do nível do mar começando com um degrau acima do nível do mar e terminando com um degrau até o nível do mar.
* Um vale é uma sequência consecutiva de degraus abaixo do nível do mar, começando com um degrau abaixo do nível do mar e terminando com um degrau até o nível do mar.

Determine a sequência de degraus para cima ou para baixo durante a caminhada, ache e escreva o número de vales atravessados.

**Exemplo:**

***steps*** = ***8 path*** = ***[DDUUUUDD]***

O caminhante entra primeiro em 2 unidades de vale profundo. Em seguida, eles escalam para fora e para 2 unidades de montanha de altura. Por fim, o caminhante retorna ao nível do mar e encerra a caminhada.

**Descrição da função:**

Complete a função ***countingValleys*** no editor abaixo.

***countingValleys*** contém o(s) seguinte(s) parâmetro(s):

* Int steps: o número de degraus da caminhada
* String path: uma string descrevendo os passos

**Retorna:**

Int: o número de vales atravessados.

**Formato da entrada:**

A primeira linha contem um inteiro ***steps***, o número de passos na caminhada.

A segunda linha contem uma única *string* de caracteres que descreve o caminho.

**Restrições:**

* 2 <= steps <= 106
* Path[i] E {U D}

**Amostra de entrada:**

8

UDDDUDUU

**Amostra de saída:**

1

**Explicação:**

Se nos representarmos – como nível do mar, degrau para cima /, e degrau para baixo \, a caminhada ficaria assim;

\_/\ \_

\ /

\/\/

O caminhante entra e sai do vale.