

---

## Número de subsecuencias felices y tristes en un string con tres secciones

X16298\_es

---

En este ejercicio tenéis que hacer varias cosas.

En primer lugar, tenéis que implementar una función que recibe un string  $s$  que cumple unas condiciones muy particulares:  $s$  está formado por tres caracteres diferentes  $c_1, c_2, c_3$ . Además, al principio de  $s$  encontramos el carácter  $c_1$  una o más veces, después viene el carácter  $c_2$  una o más veces, y finalmente viene el carácter  $c_3$  una o más veces. La función tendrá también tres parámetros enteros por referencia  $n_1, n_2, n_3$ , donde deberá asignar el número de ocurrencias de  $c_1, c_2, c_3$ , respectivamente. Esta es la cabecera:

```
// Pre: s is formed with three different characters c1, c2, c3, and is of the form
// Post: n1, n2, n3 are the number of occurrences of c1, c2, c3 in s, respectively
void numberOccurrences(const string &s, int &n1, int &n2, int &n3);
```

**Nota:** los juegos de pruebas privados de este ejercicio son grandes y están diseñados para que haga falta una implementación de coste logarítmico de `numberSubsequences`. Una implementación lenta os permitirá superar solamente los juegos de pruebas públicos y obtener la mitad de la nota.

En segundo lugar, tenéis que implementar una función que recibe un string  $s$  que cumple una de las siguientes condiciones:

1.  $s$  comienza por una o más ocurrencias de :, que vienen seguidas de una o más ocurrencias de -, que vienen seguidas por una o más ocurrencias de ), y ya no hay ningún carácter más.
2.  $s$  comienza por una o más ocurrencias de (, que vienen seguidas de una o más ocurrencias de -, que vienen seguidas por una o más ocurrencias de :, y ya no hay ningún carácter más.
3.  $s$  comienza por una o más ocurrencias de :, que vienen seguidas de una o más ocurrencias de -, que vienen seguidas por una o más ocurrencias de (, y ya no hay ningún carácter más.
4.  $s$  comienza por una o más ocurrencias de ), que vienen seguidas de una o más ocurrencias de -, que vienen seguidas por una o más ocurrencias de :, y ya no hay ningún carácter más.

En los casos (1) y (2) anteriores, la función retornará el número de subsecuencias felices de  $s$ , y en los casos (3) y (4) anteriores, la función retornará el número de subsecuencias tristes de  $s$ . Una subsecuencia feliz es una subsecuencia de tres caracteres, y donde estos tres caracteres son, o bien :-:) o bien (:-; en el orden dado. Una subsecuencia triste es una subsecuencia de tres caracteres, y donde estos tres caracteres son, o bien :-( o bien )-:, en el orden dado. Esta es la cabecera:

```
// Pre: s begins with one or more occurrences of a character c1, followed by one or more
//       occurrences of a character c2, followed by one or more occurrences of a character c3, and
//       there are no more characters in s.
```

```
//      moreover, either c1c2c3 = ":-)" or c1c2c3 = "(-:" or c1c2c3 = ":-(" or
// Post: If c1c2c3 = ":-)" or c1c2c3 = "(-:", the function returns the number of
//      If c1c2c3 = ":-(" or c1c2c3 = ")-:", the function returns the number of
int numberHappyOrSadSubsequences(const string &s);
```

La función anterior anterior deberá usar convenientemente la función `numberOccurrences` descrita al principio. En caso contrario, se invalidará la entrega.

## Observación

Sólo tenéis que enviar el procedimiento requerido; el programa principal será ignorado.

## Observación

Evaluación sobre 10 puntos:

- Solución lenta: 5 puntos.
- Solución rápida: 10 puntos.

Entendemos como solución rápida una que es correcta, de coste logarítmico y capaz de superar los juegos de pruebas públicos y privados. Entendemos como solución lenta una que no es rápida, pero es correcta y capaz de superar los juegos de pruebas públicos.

## Información del problema

Autoría: PRO1

Generación: 2026-01-25T13:55:00.365Z

© Jutge.org, 2006–2026.

<https://jutge.org>