

# Encriptación de mensajes

Tiempo máximo: 1,000 s Memoria máxima: 4096 KiB

<http://www.aceptaelreto.com/problem/statement.php?id=102>

Uno de los métodos más antiguos para codificar mensajes es el conocido como *cifrado Cesar*. Su funcionamiento es simple: cada una de las letras del mensaje original es sustituida por otra letra que se encuentra un número fijo de posiciones más adelante en el alfabeto.

Así, si utilizamos un desplazamiento de 2, las apariciones de la letra ‘a’ se sustituyen por la ‘c’, todas las apariciones de la ‘b’ por ‘d’, etc. El método tradicional comienza de nuevo al llegar al final del alfabeto, de forma que, con el desplazamiento de 2, la ‘y’ se sustituye por la ‘a’ y la ‘z’ se sustituye por la ‘b’.

Los desplazamientos también pueden ser negativos; si utilizamos un desplazamiento de -1, la ‘E’ se convertirá en ‘D’, mientras que la ‘a’ pasará a ser ‘z’.

Nuestro cifrado Cesar *no* codifica los caracteres que no sean letras anglosajonas. Así, por ejemplo, los espacios o los símbolos de puntuación no sufrirán cambio alguno.

## Entrada

La entrada está formada por un número indeterminado de casos de prueba.

Cada caso de prueba consiste en una única línea cuyo primer carácter es el código de la letra ‘p’, seguido de un mensaje codificado con el método Cesar descrito antes utilizando el desplazamiento adecuado para que la letra ‘p’ se codifique con ese primer carácter.

Los casos de prueba terminan con un mensaje codificado que, una vez traducido, contiene *exactamente* la cadena “FIN”. Cuando se lee este mensaje codificado el programa debe terminar sin generar ninguna otra salida más.

## Salida

Para cada caso de prueba, el programa indicará el número de vocales *no* acentuadas que contiene el mensaje codificado.

## Entrada de ejemplo

```
pEsta cadena esta sin codificar
pfin
qbfjpvBFJPV
ozdhntZDHNT
xXzwoziui-Um
qGJO
```

## Salida de ejemplo

```
12
1
10
10
4
```

**Autores:** Marco Antonio Gómez Martín, Pedro Pablo Gómez Martín y Patricia Díaz García.