
Repeticiones de Bases

X57448_es

Se llaman *bases* a las letras 'A', 'C', 'G' y 'T' que representan los nucleótidos Adenosina, Citosina, Guanina y Timina, con los cuales se ensambla el ADN. Se trata de hacer un programa que busca las repeticiones más largas de bases en un genoma.

El trabaja con *secciones de genoma*, que son secuencias de los caracteres 'A', 'C', 'G' y 'T' acabadas en '.'. Todos los cálculos asumen que estamos en una sección concreta. Así pues, para cada base, hay que dar como resultado la longitud del tramo de repeticiones consecutivas más largo encontrado, y en qué posición está. La ocurrencia de una sola base suelta ya se considera una "repetición" de longitud 1. Cuando, para la misma base, hay dos tramos con repeticiones de la misma longitud, hay que reportar siempre el primero.

Por ejemplo, si la sección fuese "AAACCATGGGTTGTT.", el resultado debería ser el siguiente:

- para 'A', la repetición más larga es de 3, empezando en la posición 1;
- para 'C', es de 2, empezando en la posición 4;
- para 'G', es de 3, empezando en la posición 7; y
- para 'T', es de 2, empezando en la posición 11 (hay otra de longitud 2 en la posición 14, pero ésta es posterior a la posición 11)

Las posiciones se refieren siempre a una sola sección de genoma, no al genoma entero, y empiezan por 1.

Entrada

La entrada es una secuencia de caracteres donde solo aparecen 'A', 'C', 'G', 'T', y '.', descontando saltos de línea. Cada sección de genoma es una secuencia de bases acabada en '.' y la entrada entera es una secuencia de secciones.

Salida

A la salida hay que mostrar los resultados de cada sección, por orden y en una sola línea, y con el siguiente formato:

```
"A" = 0, "C" = 2(1), "G" = 0, "T" = 4(3)
```

Para cada base hay que mostrar la base entre comillas, un signo de igual flanqueado por espacios y luego la longitud máxima encontrada. Si ésta es mayor que 0, entonces hay que añadir la posición entre paréntesis. Las bases están ordenadas alfabéticamente y se separan entre ellas con una coma y un espacio.

Ejemplo de entrada 1

```
ACGT.AACCGGTT.AAACCCGGGTTT.  
AAAACCCCGGGGTTTT.AAAACCCGGT.  
ACCGGTTTT.
```

Ejemplo de salida 1

```
"A" = 1(1), "C" = 1(2), "G" = 1(3), "T" = 1(4)  
"A" = 2(1), "C" = 2(3), "G" = 2(5), "T" = 2(7)  
"A" = 3(1), "C" = 3(4), "G" = 3(7), "T" = 3(10)  
"A" = 4(1), "C" = 4(5), "G" = 4(9), "T" = 4(13)  
"A" = 4(1), "C" = 3(5), "G" = 2(8), "T" = 1(10)  
"A" = 1(1), "C" = 2(2), "G" = 3(4), "T" = 4(7)
```

Ejemplo de entrada 2

.AAA.GGAG.C.CTTTTTCCTTT.

Ejemplo de salida 2

```
"A" = 0, "C" = 0, "G" = 0, "T" = 0
"A" = 3 (1), "C" = 0, "G" = 0, "T" = 0
"A" = 1 (3), "C" = 0, "G" = 2 (1), "T" = 0
"A" = 0, "C" = 1 (1), "G" = 0, "T" = 0
"A" = 0, "C" = 2 (1), "G" = 0, "T" = 4 (3)
```

Información del problema

Autoría: PRO1

Generación: 2026-01-25T17:03:13.688Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>