

---

**Repeticions de Bases****X57448\_ca**

---

S'anomenen *bases* les lletres 'A', 'C', 'G' i 'T' que representen els nucleòtids Adenosina, Citosina, Guanina i Timina, amb els quals es construeix l'ADN. Es tracta de fer un programa que busca les repeticions més llargues de bases en un genoma.

El programa treballa amb seccions del genoma, és a dir, seqüències dels caràcters 'A', 'C', 'G' i 'T' acabades amb '.'. Tots els càlculs fan la suposició que estem en una secció concreta. Així doncs, per a cada base, cal mostrar el tram de repeticions consecutives més llarg trobat, i en quina posició apareix. La ocurrència d'una sola base aïllada ja es considera una "repetició" de longitud 1. Quan, per a la mateixa base, hi ha dos trams amb repeticions de la mateixa longitud, cal reportar sempre el primer.

Per exemple, si la seqüència fos "AAACCATGGGTTGTT.", el resultat seria el següent:

- per a 'A', la repetició més llarga és de 3, començant a la posició 1;
- per a 'C', és de 2, començant a la posició 4;
- per a 'G', és de 3, començant a la posició 7; i
- per a 'T', és de 2, començant a la posició 11 (n'hi ha una altra de longitud 2 començant a la posició 14, però aquesta posició és posterior a la posició 11).

Les posicions es refereixen sempre a una sola secció de genoma, no al genoma sencer, i comencen per 1.

**Entrada**

L'entrada és una seqüència de caràcters a on només hi ha 'A', 'C', 'G', 'T', i '.', sense tenir en compte els salts de línia. Cada secció de genoma és una seqüència de bases acabada en '.' i l'entrada sencera és una seqüència de seccions.

**Sortida**

A la sortida s'han de mostrar els resultats de cada secció, per ordre i en una sola línia, en el següent format:

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

Per a cada base cal mostrar la base entre cometes dobles, un signe d'igual entre dos espais i després la longitud màxima trobada. Si aquesta és major que 0, llavors cal afegir la posició entre parèntesis. Les bases estan ordenades alfabèticament i se separen entre elles amb una coma i un espai.

**Exemple d'entrada 1**

```
ACGT.AACCGGTT.AAACCGGGTTT.  
AAAACCCCGGGGTTTT.AAAACCGGT.  
ACCGGGTTTT.
```

**Exemple de sortida 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)
```

## Exemple d'entrada 2

.AAA.GGAG.C.CCTTTTCCTTT.

## Exemple de sortida 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)

## Informació del problema

Autoria: PRO1

Generació: 2026-01-25T17:03:17.738Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>