Techniques de Programmation
TP1 : Chaînes de caractères

Olivier Hoenen – `hoenen@icps.u-strasbg.fr`

16 février 2007

Écrire des petites fonctions C pour répondre aux questions qui suivent. On pourra s'aider des fonctions prédéfinies `strcpy`, `strlen` et `strcmp` (ne pas oublier d'inclure `string.h`).

Parcours de base

1. Rentrer une séquence d'ADN au clavier et la stocker dans une chaîne de caractères. Cette chaîne sera la *séquence de travail.*
2. Vérifier que la séquence de travail est composée uniquement de caractères 'A', 'C', 'G', et 'T'.
3. Calculer le nombre d'occurences de chacun de ces caractères et afficher cet histogramme à l'écran.
4. Un palindrome est un mot ou une phrase qui se lit indifféremment en commençant au début ou à la fin. Tester une chaîne de caractère pour déterminer si il s'agit d'un palindrome ou non.

Recherche de motifs

1. Rentrer un *codon* au clavier, soit une séquence de 3 éléments (3 bases).
2. Tester la présence du codon dans la séquence. Afficher ses positions successives dans la séquence ainsi que le nombre total d'occurences.
3. Calculer la plus grande *plage* (répétitions successives) d'une base dans une séquence donnée. Afficher la position du début de cette plage dans la séquence.
4. Généraliser la fonction précédente en remplaçant la base recherchée par une séquence.

Conversions

1. Écrire le complémentaire de la séquence ADN de travail ($A \leftrightarrow T$ et $G \leftrightarrow C$). Attention à ne pas modifier la séquence de départ.
2. Convertir une séquence ADN en ARN ($T \rightarrow U$). Modifiez cette fois la séquence passée en argument.
3. Tester avec la séquence de travail puis avec une séquence déclarée comme ceci : `char *seqc = ‘‘ATGCGAGAGTGCGA’’;`