UNIVERSITÉ DE MONTPELLIER
PLACE EUGÈNE BATAILLON 34095 MONTPELLIER CEDEX 5
Master Bioinformatique – 2023/2024

Développement opérationnel avancé
Travaux Pratiques n° 1

Réalisation d'une API de manipulation des fichiers FASTA

Alban MANCHERON

1 Description du format FASTA

Le format FASTA permet de représenter les séquences nucléiques. Ce format est utilisé par l'ensemble de la communauté des biologistes et bioinformaticiens. Ce format fût initialement utilisé par le logiciel éponyme développé par D. LIPMAN & W. PEARSON en 1985.

Une séquence au format FASTA débute par une ligne de description commençant par le symbole '>' (historiquement, la ligne de description pouvait également débuter par ';', voire même être écrite sur plusieurs lignes successives commençant par ';'). Les lignes suivantes décrivent la séquence biologique (nucléique ou protéique) en suivant le standard IUPAC à une lettre, jusqu'à rencontrer la fin du fichier ou la description d'une nouvelle séquence. L'usage recommande de décrire la séquence biologique par blocs de lignes de 60 à 80 symboles. Ce n'est qu'un usage et en aucun cas une obligation.

Exemple de séquence au format FASTA :

```
>gi|295656579|gb|HM026173.1| Homo sapiens isolate W97X sex determining region Y gene, partial cds
GCGAAACTCAGAGATCAGCAAGCAGCTGGGATACCAGTGAAAAATGCTTACTGAAGCCGAAAAATGGCCA
TTCTTCCAGGAGGCACAGAAATTACAGGCCATGCACA
```

Étant devenu un standard d'échange des données, la ligne descriptive de la séquence est très souvent une suite d'informations séparées par des '|' sans espaces (*c.f.* recommandations du NCBI). Il arrive parfois que la communauté utilise le terme de fichier multi-FASTA pour distinguer les fichiers contenant plusieurs séquences des fichiers n'en contenant qu'une seule.

Vous trouverez de plus amples informations sur le site de Wikipedia [1], ainsi que sur le site du NCBI [2].

2 Objectif du TP

Votre travail consiste dans un premier temps à réaliser une API (*Application Programming Interface*) permettant de manipuler les fichiers FASTA et plus généralement les séquences biologiques.

Vous devez donc permettre de lire un fichier dans ce format, vérifier sa conformité (en affichant au besoin les recommandations non respectées et les erreurs rencontrées afin d'assister un éventuel utilisateur dans la correction de ses fichiers).

Votre API doit permettre le comptage du nombre de séquences présentes dans le fichier, d'analyser le ou les entêtes des séquences (ou d'une séquence en particulier), de récupérer la ou les séquences dans un objet informatique facile à manipuler (calcul des séquences complémentaires inversées, extraction de sous-séquences, ...).

Bien évidemment, il vous est recommandé de trouver des solutions **efficaces** en temps comme en espace pour répondre à ce cahier des charges.

1. http://en.wikipedia.org/
2. http://www.ncbi.nlm.nih.gov/

Enfin, eu égard aux volumes de données biologique sans cesse croissant, la plupart des fichiers sont échangés après compression (`gzip`, `bzip2`, ...). Nous vous proposons donc, en option, d'intégrer à votre API la gestion de fichiers compressés avec `gzip` en les lisant « à la volée ». Cette possibilité n'étant pas primordiale, il vous est suggéré de ne l'intégrer que si les demandes préalables sont entièrement satisfaites.

3 Informations complémentaires

Pour mener à bien votre développement, il vous est conseillé de fortement vous documenter au préalable (*e.g.*, sur la norme IUPAC ; sur la manière dont vous coderez vos séquences nucléiques – et donc la manière dont vous calculerez les opérations de complémentaire inversé ; sur les recommandations du NCBI pour les entêtes FASTA ; ...).

Pour chaque méthode que vous développerez, vous calculerez l'ordre de grandeur de sa complexité (temps et espace). Vous vérifierez que les temps de calculs et la mémoire utilisée [3] observés sur différents jeux d'essais sont cohérents avec la complexité temporelle et spatiale calculée...

Il vous est également recommandé de proposer plusieurs méthodes donnant le même résultat (pensez à exploiter le polymorphisme offert par les langages objets comme `C++`).

Pour tester votre API, vous devrez récupérer des données sur le site du NCBI. Il vous est demandé de récupérer des séquences de différentes longueurs ($\simeq 100pb$, $\simeq 1\ 000pb$, $\simeq 1\ 000\ 000pb$).

Enfin, sachez que ce travail constitue les prémices du projet. Il vous est donc fortement recommandé de vous y consacrer sérieusement.

Bon Courage...

3. En `C++` (sous GNU/Linux), vous pourrez utiliser la fonction `getrusage(int, struct rusage *)` déclarée dans `<sys/resource.h>`.