Polytech
Biotech III

Module : Bioinformatique et analyse de données.
Responsables : D. Puthier, A. Sergé, N. Terrapon.
Date : Mardi 24 Janvier 2017.
Durée : 2 heures.
Calculatrices/portables : non autorisés.
Documents : non-autorisés.

1. Vous répondrez aux questions qui suivent sur la feuille d'examen qui vous sera fournie.
2. Ecrivez lisiblement.
3. Respectez bien les consignes qui vous sont données dans chacune des questions.
4. N'oubliez pas de signer la feuille d'émargement après avoir rendu votre copie.

---

1 Introduction

Nous ferons référence, par la suite, au fichier *transcript.fasta*. Ce fichier contient **les séquences correspondant aux exons de transcrits d'un génome fictif)**. Comme tout fichier FASTA il contient une suite d'enregistrements composés comme suit :

- Une ligne pour **l'entête ('header' pour les anglo-saxons)**, commençant par le caractère ">" (un chevron) et qui contient des informations permettant de qualifier la séquence qui se trouve à la ligne suivante. Ici **l'entête contient le nom du transcrit, le numéro de l'exon et le brin** (le caractère 'tube' ('|') est utilisé comme séparateur,).
  ☞ `>Nom du transcrit|numero de l'exon|brin`
- **Une ligne pour la séquence nucléotidique de l'exon**. Notez, qu'ici les points de suspension indiquent que les séquences ne sont pas données dans leur totalité.
  ☞ `ATCGATAACATACAAGCTA...`

On donne ci dessous les premières lignes du fichier *transcript.fasta*. On peut y voir les lignes correspondant à trois transcripts (A, B et C) contenant respectivement 3, 2 et 4 exons/séquences. Deux d'entre eux (A et B) sont produits par le brin + tandis que C est produit par le brin -.

```
>A|1|+
GATAGATATAAATATCAAATCATC
>A|2|+
ACTAAAACACACACACCAACACCA
>A|3|+
ACGACAACTAGCACTAAGATCACA
>B|1|+
CAGCAACAAGATAAAGCCGGGGAT
>B|2|+
TCAGTAAGTAAAGATAAAGAAGAT
>C|1|-
TCAGTAAGTAAAGATAAAGAAGAT
>C|2|-
TATCAGACGAGACAGACAGCAGAG
>C|3|-
CTACCGCAGCATACAGACAACCAC
>C|4|-
GGTGCTGCCTCGCCTCGCTCGCTC
...
```

2 Commandes Unix : "Première machine sous Unix" (4 points)

D. Ritchie, est étudiant, débutant dans le milieu de la bioinformatique. D. Ritchie, viens de recevoir son nouvel ordinateur. L'un de ses collègue l'a aidé à installer un système Unix. Sur ce système, son répertoire utilisateur ('home') se trouve dans */home/ritchie*. Il se connecte, ouvre un terminal et cherche à faire ses premiers pas.

Questions : Vous devez l'aider à répondre aux questions suivantes à l'aide d'une commande Unix.

1. Comment se déplacer dans son répertoire home.
2. Comment lister le contenu du répertoire courant (*i.e.* dans lequel il se trouve).
3. Comment télécharger le fichier *transcript.fasta.gz* à l'adresse *http ://ritchie-user.fr/transcript.fasta.gz*
4. Comment décompresser le fichier *transcript.fasta.gz* (on obtiendra un fichier sans l'extension .gz).

5. Comment compter les lignes du fichier transcript.fasta.
6. Comment regarder les 20 premières lignes du fichier transcript.fasta.
7. Comment regarder les 20 dernières lignes du fichier transcript.fasta.
8. Comment, en utilisant grep, compter le nombre de d'enregistrements (i.e de lignes contenant ">").
9. Comment créer un dossier *result* dans son dossier 'home' ?
10. Comment renommer le dossier *result* en *results* (faute de frappe...) ?
11. Comment déplacer le fichier *transcript.fasta* dans le dossier *results*.

3 Expressions régulières : "Cherchons des motifs" (4 points)

Maintenant que notre étudiant à configuré sa machine, son premier travail est d'analyser le contenu du fichier fasta qu'on lui a fourni (transcript.fasta). Son responsable lui a soumis une liste de questions auxquelles il souhaiterait pouvoir répondre. Pour répondre à ces questions il aura besoin d'utiliser la commande *grep* pour trouver les lignes vérifiant des expressions régulières. La forme des commandes sera celle indiquée ci-dessous (les points de suspensions seront remplacés par une expression régulière).

```
d.ritchie@machine: grep ...  transcript.fasta
```

Le tableau ci-dessous contient quelques opérateurs d'expressions régulières et leurs significations que vous pourriez être appelés à utiliser :

Opérateur	**Signification**
.	un caractère quelconque.
[a-z]	Une lettre minuscule (interval, ex : [u ? w]).
[A-Z]	Une lettre majuscule (interval, ex : [A ? E]).
[ABc]	A ou B ou c.
[^ABab]	Toute lettre différente de a et b (minuscule ou majuscule).
^	Début de ligne.
$	Fin de ligne
+	1 ou n fois le caractère qui précède.
*	0 à n fois le caractère qui précède.
n,m	Le caractère qui précède répété entre n et m fois.
\	Caractère d ?échappement.

Questions : Construisez les expressions régulières permettant de sélectionner les lignes suivantes.

1. Les lignes contenant le caractère ">" ?
2. Les lignes contenant le motif (*i.e* suite de caractères) 'ATGCC' ?
3. Les lignes contenant le motif 'atgcc' ?
4. Les lignes contenant le motif 'ATC' suivi de 2 à 3 caractères suivis du motif ATG ?
5. Les lignes commençant par 'AGG' ?
6. Les lignes finissant par 'AAA' ?
7. Les lignes commençant par 'AGG' et finissant par 'AAA' ?
8. Les lignes commençant par 'AGG', finissant par 'AAA' et contenant le motif 'GGG' entre les deux ?
9. Les lignes ne contenant que les caractères A, T, G ou C.
10. Les lignes lignes vides ?
11. Les lignes ne contenant pas les caractères A, T, G ou C ?

4 Python : "Mince alors" (4 points)

D. Ritchie a écrit un programme assez rudimentaire permettant de convertir le fichier *transcript.fasta* en un format tabulé dont la forme est la suivante :

```
transcript      exon_number     strand  sequence
A       1       +       GATAGATATAAATATCAAATCATC
A       2       +       ACTAAAACACACACACCAACACCA
...
```

Malheureusement, à la suite d'une erreur de manipulation (avec la commande gshuf), il a qui a randomisé les lignes de son programme. Zut...

Question : réorganisez les lignes du programme pour que celui-ci puisse à nouveau faire ce pour quoi il a été écrit. Reportez le code réorganisé dans votre copie d'examen.

```
01          header =  transcript + "\t" +  exon_number + "\t" + strand
02          print("transcript\texon_number\tstrand\tsequence")
03      if line.startswith(">"):
04      n += 1          line = line.lstrip(">")
```

```
05 for line in file_handler:
06         fields = line.split("|")
07     line = line.rstrip("\n")
08         print(header + "\t" + line)
09 file_handler = open("transcript.fasta")
10         exon_number = fields[1]
11     if n == 0:
12         strand = fields[2]
13         transcript = fields[0]
14 n=0
15     else:
```

5 Python : "Où sont les commentaires !" (5 points)

D. Ritchie a écrit un programme mais, en l'absence de commentaires, il ne sait plus ce qu'il était censé faire. Le code du programme est donné ci-dessous. Recopiez le code sur votre copie en le commentant et en utilisant des noms de variables plus explicites (3 points). **Question** : Qu'est censé faire ce programme ? Expliquez. (2 points).

```
01        fh = open("transcript.fasta")
02        nb = dict()
03        for l in fh:
04            l = l.rstrip("\n")
05            if l.startswith(">"):
06                l = l.lstrip(">")
07                f = l.split("|")
08                t = f[0]
09                if t not in nb:
10                    nb[t] =1
11                else:
12                    nb[t] +=1
13        fh.close()
14        fh = open("transcript.fasta")
15        print(nb)
16        for l in fh:
17            l = l.rstrip("\n")
18            if l.startswith(">"):
19                h = l
20                l = l.lstrip(">")
21                f = l.split("|")
22                t = f[0]
23            else:
24                if nb[t] >= 4 :
25                    print(h)
26                    print(l)
```

6 Python "Sélection aléatoire" (3points)

Questions : Ecrivez un programme permettant de lire le fichier "transcript.fasta" et de sélectionner au hasard 10 transcripts dont les séquences des exons correspondant devront être imprimées au format FASTA.
Vous pouvez dans ce programme utiliser la fonction randint() du module random. Celle-ci permet de tirer, au hasard, un entier dans un intervalle [n, m] (exemple ci-dessous).

```
from random import randint
n=2
m=5
randint(n,m) # un entier au hasard parmi: 2,3,4,5
```