DS Bioperl TP

Utiliser Perl et la bibliothèque Bioperl pour chacune des questions. Utiliser les « use » que vous voulez. Les exercices sont indépendants les uns des autres sauf les questions 9 et 10. Les données pour répondre aux exercices sont dans le dossier « Donnees_TP ». Durée du TP : 1 H 30

1) En utilisant Bioperl créer une séquence, afficher la séquence, la traduire en protéine et afficher la séquence d'acides aminés.
Estimation : 3 minutes – 0.5 point

2) Lire le fichier sequences_1.fasta contenant plusieurs séquences puis afficher une liste des noms des séquences, leur longueur et les 20 premières bases.
Estimation : 4 minutes – 0.75 point

3)

```
#!/usr/bin/perl
use strict;
use warnings;
use Bio::Seq;
use Bio::SeqIO;
my ($seq, $seqinv, $out, $i);
#Création d'un objet sequence
$seq=Bio::Seq->new(-id =>'testseq',-seq => 'CATGTAGATAG');
```

Compléter en affichant à l'écran la traduction de la séquence suivant les 6 phases de lecture.
Estimation : 8 minutes – 1.75 points

4) Écrire un programme nommé gen2fasta.pl qui permet de convertir un fichier au format Genbank en fichier fasta. Les noms des fichiers de départ et d'arrivée devront être des arguments de la ligne de commande. Tester ce programme sur le fichier Genbank "sequence.gb".
Estimation : 4 minutes – 0.75 point

5) Lire le fichier Homo_sapiens.GRCh38.cdna.all.fa et afficher la description et les 5 premières bases de la séquence portant l'identifiant « ENST00000523715.2 »
Estimation : 4 minutes – 0.75 point

6) Lire la liste des numéros d'accession GenBank contenus dans le fichier liste_seq, récupérer les données correspondantes dans le fichier liste_seq.gb et les écrire dans un fichier fasta.
Estimation : 10 minutes – 2.5 points

7) Créer une séquence et afficher le nombre de chaque monomère.
Estimation : 5 minutes – 1 point

8) Récupérer un objet qui représente l'enzyme EcoRI et afficher son nom et son site de reconnaissance.
Estimation : 3 minutes – 0.5 point

9) Couper toutes les séquences contenues dans sequences_2.fasta avec des enzymes qui reconnaissent une séquence de 6 bases. Afficher le nom de la séquence, le nom de l'enzyme avec son site de reconnaissance et les fragments générés.
Extrait des résultat à obtenir :
>JX220971.1
>>AasI GACNNNN^NNGTC
>>>1
GAAAAAAAGAAA...
>>>2
ATGTCGCAT...
>>AatI AGG^CCT
>>>1
GAAAAAAAGAAA…
Afin d'obtenir les enzymes d'une collection il faut utiliser sur une collection la méthode :
-> each_enzyme qui renvoie un tableau d'enzyme.
Exemple :
my @enzymes=$all_collection -> each_enzyme ;
($all_collection correspond à une collection d'enzyme de classe Bio::Restriction::EnzymeCollection)
Estimation : 15 minutes – 3.25 points

10) En reprenant votre script précédant. Couper toutes les séquences contenues dans sequences_2.fasta avec des enzymes qui reconnaissent une séquences de 6 bases en ajoutant une enzyme « Toto » avec son site de reconnaissance « ATA^CAA ».
Estimation : 4 minutes – 0.75 point

11) Afficher le nom de la séquence et les 3 meilleurs propositions de sites de clivage donnés par SigCleave pour chaque séquence du fichier sequences_2.fasta.
Estimation : 10 minutes – 2.5 points

12) Faire un BLAST entre les séquences requêtes (= query) du fichier liste.gb contre la base de données « bdseq.txt ». Puis, afficher le premier hit (si il y en a un) en lisant le fichier résultat de BLAST (par défaut le format de ce fichier est "blast") pour chacune des séquences requêtes.
Affichage voulu :
>AF068625
deshydrogenase
>NM_001130823
Estimation : 10 minutes – 2.5 points

13) Ecrire un programme qui lit le fichier Amplicons_echantillon_9.fasta en format fasta et qui l'imprime en format genbank tout en ajoutant les annotations "freshwater" pour le tag "origin" pour chacune des séquences. Aussi, ajouter la feature suivante sur la première séquence seulement :
FEATURES Location/Qualifiers
HIT 1..362
/description="NR_134232.1"
Estimation : 10 minutes – 2.5 points