Séance 6 / BIBS 2024

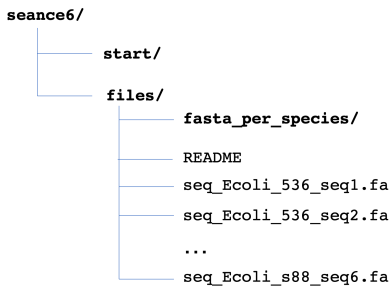
interaction avec son environnement / manipulation de fichiers / répertoires (dirs)

**** *Parcours de répertoires*

1/ Créer un répertoire "seance6".

2/ Télécharger l'archive files.tar sur ecampus qui contient un répertoire contenant des fichiers fasta (extension ".fa"). Stocker ce répertoire et non pas uniquement les fichiers dans un répertoire "seance6".

3/ Créer un répertoire "start" et se placer dans "start" pour commencer la séance. Vous devez avoir cette architecture :



4/ Ouvrir un script python et à partir de ce script réaliser les tâches suivantes :

5/ Afficher le chemin du répertoire "start". Lister le contenu des fichiers et dirs du répertoire où vous vous trouvez. Se déplacer dans le répertoire "files" et lister son contenu en utilisant le module os. Vous constaterez que le répertoire contient des fichiers fasta contenant des séquences au format fasta. On rappelle qu'au format fasta, chaque séquence est matérialisée par une ligne commençant par un ">" suivi de son identifiant et les lignes correspondant à sa séquence. Autrement dit, un fichier fasta contient autant de séquences que de ">". Ces fichiers contiennent des séquences de Ecoli provenant de différentes souches. Le nom de chaque souche est renseigné dans le nom de fichier : seq_Ecoli_536_seq1.fa contient une ou plusieurs séquences de la souche Ecoli_536.

On se propose de générer un fichier texte "report_fasta.txt" qui contiendra toutes les infos relatives à chaque fichier fasta contenus dans "files". Nous allons le construire au fil de la séance et il suivra ce format :

```
#summary
nb_items = 25
nb_files = 24
nb_fasta = 23
nb_dir = 1

#details for each fasta file
# filename                          # size     #nb_lines     #nb_seq
seq_Ecoli_536_seq1.fa                 324B          8            1
         ...
```

6/ Utiliser le module os pour identifier les items correspondant à des fichiers, à des répertoires et compter le nombre d'items dans chaque catégorie. Tester si le fichier " seq_Ecoli_536A_seq1.fa" existe.

**** *Manipulation automatique de fichiers*

7/ Utiliser le module glob pour compter le nombre de fichiers fasta. Vous remarquerez que les fichiers fasta se terminent ici soit par l'extension ".fa", ".fasta" ou ".fsa". Vous avez alors toutes les informations pour remplir les lignes "summary" de votre fichier " report_fasta.txt".

8/ Grâce à os.system(), créer un répertoire "copie/" et copier tous les fichiers fasta dedans. Rappel d'utilisation de la commande "cp" :

```
> cp   file1.txt   dir/      # copie le fichier "file1" dans le répertoire dir/
> cp   file*.txt   dir/      # copie tous les fichiers file*.txt dans le répertoire dir/
```

9/ Grâce à os.system(), renommer les extensions de tous les fichiers fasta du répertoire "files" par l'extension ".fasta". Rappel d'utilisation de la commande "mv" :

```
> mv   file1.txt   file1.texte  # renomme le fichier file1.txt en file1.texte
```

Si la commande se "passait mal", vous pouvez récupérer vos fichiers originaux dans le répertoire "copie".

10/ Ecrire une fonction qui calcule pour chaque fichier, son nombre de lignes ainsi que le nombre de séquences qu'il contient.

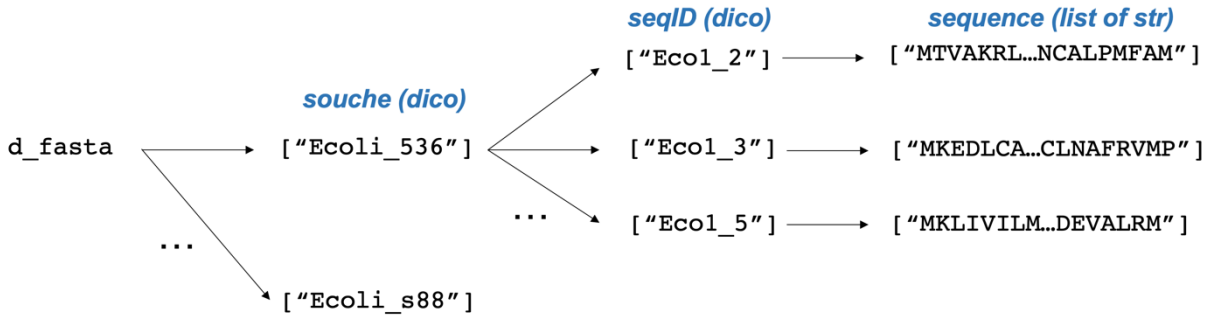
11/ Calculer le nombre de lignes, de séquences et la taille de chacun des fichiers fasta et remplir le "report_fasta.txt".

12/ On veut maintenant regrouper les séquences appartenant à une même souche dans un même fichier que l'on nommera nomSouche_sequences.fasta et qu'on stockera dans le répertoire "fasta_per_species/". Par exemple, on regroupera toutes les séquences de la souche "Ecoli_536" dans un même fichier qu'on nommera "Ecoli_536_sequences.fasta" qui sera stocké dans le répertoire "fasta_per_species". Créer les commandes qui donc, automatiquement, regrouperont toutes les séquences d'une même souche dans un même fichier et qui stockera le fichier résultant dans le répertoire "fasta_per_species/".

13/ Se déplacer dans le répertoire "fasta_per_species/" et adapter le script précédent pour créer un nouveau fichier "report_fasta.txt" qui recensera le nombre de nouveaux fichiers fasta, leur taille, nb de lignes et nb de séquences.x

**** *Analyse des fichiers fasta*

14/ Créer un parser de fichier fasta qui stockera les informations de séquences dans un dictionnaire comme suit :



15/ Créer une fonction qui va parcourir chaque séquence et l'annoter "codante" ou "non-codante" en fonction des règles suivantes : "codante" ssi la séquence contient plus de 60 acides-aminés et commence par une Methionine. Votre fonction ajoutera deux nouvelles clés ["length"] et ["status"] à la clé ***seqID*** de votre dictionnaire d_fasta (çàd : au mini dictionnaire correspondant à chaque séquence (d_fasta["Ecoli_536"]["Eco1_2"] par exemple)) et les remplira comme suit :

- d_fasta["Ecoli_536"]["Eco1_2"] ["length"] = 124 # longueur de la séquence "Eco1_2"
- d_fasta["Ecoli_536"]["Eco1_2"] ["status"] = "coding" ou "noncoding" en fonction de ses propriétés

16/ Créer une fonction qui prendra en argument une liste contenant une séquence, comptera le nombre de chacun des 20 acides aminés et renverra un dictionnaire de 20 clés correspondant aux 20 acides aminés possibles. Chaque clé pointera vers le nombre de fois où cet acide aminé a été observé dans la séquence. Votre programme ajoutera ensuite ce nouveau dictionnaire la clé ***seqID*** de votre dictionnaire d_fasta. Votre dictionnaire devra ressembler à ça.