

# UE SBIAU04L - Introduction à l' informatique et la programmation (2022)

## Projet de programmation Python

### Présentation du projet

L'objectif de ce projet est de développer un "traducteur" permettant de convertir des séquences d'ADN en séquences d'ARN, puis en séquences d'acides aminés. Ce traducteur, codé en python, doit pouvoir :

- Être utilisé en lignes de commandes ou avec une interface graphique, selon le choix de l'utilisateur. Ce choix doit être fait au lancement du programme.
- Proposer les différents niveaux de conversion : ADN > ARN, ARN > protéine, ADN > protéine, selon le choix de l'utilisateur.
- L'utilisateur doit pouvoir fournir une ou plusieurs séquence(s) de son choix à transcrire/traduire, ou un fichier au format FASTA contenant plusieurs séquences.
- L'utilisateur doit également pouvoir fournir un fichier FASTA avec une longue séquence nucléique (ex. un chromosome) accompagné d'un fichier GTF/GFF contenant les positions des gènes d'intérêt à transcrire et/ou traduire.
- Si le format de la séquence attendue (ex. ADN ou ARN) ne correspond pas à celle fournie (ex. ARN), alors une erreur doit être reportée
- Si un codon stop apparaît, alors la traduction doit s'arrêter.
- L'utilisateur peut fournir une table d'utilisation des codons alternative des standards.

A vous de créer des sous-fichiers tests, notamment pour générer les erreurs potentielles d'un utilisateur « beta », essentiels pour démontrer le bon fonctionnement de votre code.

### Partie 1

Créez deux fonctions, selon le choix de l'utilisateur :

- Une fonction split() qui va lire votre fichier gtf/gff et extraire les séquences d'intérêt au format FASTA.
- Une fonction fasta(), qui lit un fichier FASTA.

### Partie 2

Créer une fonction control() qui va vérifier que le format des données entrées (FASTA, gtf/gff) est correct, sinon retourne un message d'erreur.

### Partie 3

Créez les fonctions transcription(), et traduction() qui permettent de passer d'ADN à ARN, et d'ARN à protéine.

Note : aidez-vous de dictionnaires.

#### **Partie 4**

Créez la fonction `main()` permettant de faire la transcription, la traduction, ou les deux en fonction de choix de l'utilisateur.

#### **Partie 5**

Créez une fonction `erreur()`, permettant de retourner différents messages d'erreur si la séquence ne correspond pas au format attendu selon le choix de l'utilisateur.

#### **Partie 6**

Créez une interface graphique dans laquelle l'utilisateur peut donner une séquence ou donner un fichier FASTA, et décider quelle(s) opération(s) faire.