

## TP ALGO 9 : LECTURE ET ÉCRITURE DE FICHIERS .TXT ET REPRÉSENTATIONS GRAPHIQUES DE JEUX DE DONNÉES

### PREMIÈRES MANIPULATIONS DE FICHIERS

- Télécharger tous les fichiers d'extension `.txt` sur ma page web (il y en a 4 télécharger, éventuellement à l'aide d'un clic-droit) et les enregistrer dans un dossier.
  - Ouvrir (double clic) dans un éditeur de texte les fichiers `donnees1.txt` et `donnees2.txt`, observer leur contenu puis les fermer.
  - Créer un script vide nommé TP9, l'enregistrer dans le même dossier que les fichiers `.txt` précédemment téléchargés et l'exécuter.  
*Cette manipulation est essentielle car elle permet artificiellement de se placer dans le répertoire où se trouvent les fichiers : si vous sautez cette étape, une erreur vous indiquera que le logiciel ne trouve pas les fichiers !*

- Taper dans le Shell :

```
>>> fichier1 = open('donnees1.txt','r')
>>> contenu1 = fichier1.read()
>>> fichier1.close()
>>> contenu1
```

```
>>> fichier2 = open('donnees2.txt','r')
>>> contenu2a = fichier2.readline()
>>> contenu2b = fichier2.readline()
>>> fichier2.close()
>>> contenu2a
>>> contenu2b
>>> type(contenu2b)
>>> contenu2b[0]
```

```
>>> contenu2bBis = contenu2b.split(' ')
>>> contenu2bBis
>>> type(contenu2bBis)
>>> contenu2bBis[0]
>>> eval(contenu2bBis[0])
```

- Taper dans le Shell :

```
>>> fichier3 = open('nouveau.txt','w')
>>> message1 = 'Résultat : '
>>> fichier3.write(message1)
>>> fichier3.close()
```

Aller voir dans le dossier où sont vos fichiers et vérifier la présence (et le contenu) du fichier `nouveau.txt`.

- Taper dans le Shell :

```
>>> res = 245
>>> fichier4 = open('nouveau.txt','a')
>>> message2 = str(res)+'\n'+ 'Fin'
```

```
>>> fichier4.write(message2)
>>> fichier4.close()
```

Aller de nouveau vérifier le contenu du fichier `nouveau.txt`.

## SÉQUENCES ADN

5. Un brin d'ADN peut se représenter par un mot (en langage Python : une chaîne de caractère) créé à partir d'un alphabet codant des molécules ordonnées dans un certain ordre (correspondant aux nucléotides). Dans les exemples qui suivent, les molécules considérées seront l'adénine (A), la cytosine (C), la guanine (G) et la thymine (T).

Exemple : AGCTTTTCATTCTGACTGCAACGGGCAATATGTC

- (a) Ouvrir (double clic) dans un éditeur de texte le fichier `hamming.txt`, observer son contenu (*il contient deux lignes, chacune comportant une chaîne de caractères correspondant à un brin d'ADN*) puis le fermer.
- (b) Dans l'éditeur, écrire un programme permettant d'ouvrir le fichier `hamming.txt` puis de stocker son contenu dans deux variables de type chaîne de caractères qu'on appellera `brin1` et `brin2`.
- (c) Écrire à la suite du programme une fonction `repartition` qui prend un argument une chaîne de caractères `brin` et qui compte et renvoie le nombre de *A*, le nombre de *C*, le nombre de *G* et le nombre de *T* composant `brin`.

```
>>> repartition(brin1) .....
```

```
>>> repartition(brin2) .....
```

- (d) La « teneur en GC » d'un brin d'ADN est le pourcentage de symboles *G* et *C* dans la chaîne de caractère correspondante. Écrire une fonction `GCcontent` qui prend en argument une chaîne de caractères `brin` et qui renvoie la teneur en GC de `brin`.

```
>>> GCcontent(brin1) .....
```

```
>>> GCcontent(brin2) .....
```

- (e) Une mutation est une erreur qui se produit lors de la création ou de la copie d'un brin d'ADN. La mutation la plus simple que l'on peut considérer est celle où une base est remplacée par une autre. On cherche à déterminer le nombre de mutations différenciant deux brins d'ADN. Étant donné deux chaînes de caractères `s` et `t` de même longueur, la *distance de Hamming* entre `s` et `t`, notée  $d_H(s, t)$ , correspond au nombre de lettres qui diffèrent dans `s` et `t`.

Exemple : si  $s = \text{GAGCCTA}$  et  $t = \text{CATCGTA}$ ,  $d_H(s, t) = 3$ .

Écrire dans le même script une fonction `Hamming` en langage Python qui prend en arguments deux chaînes de caractères `s` et `t` et qui renvoie la distance de Hamming  $d_H(s, t)$ .

```
>>> Hamming(brin1,brin2) .....
```

6. On souhaite générer un fichier texte résumant les résultats obtenus aux questions précédentes. A l'aide de la fiche de cours, écrire une fonction `compteRendu` qui étant donné deux chaînes de caractères `s` et `t` de même longueur

- ★ crée un fichier texte `CR.txt`,
- ★ calcule la teneur en GC de chaque chaîne et la distance de Hamming entre les deux chaînes,
- ★ écrit les résultats dans le fichier `CR.txt` sous le format suivant :

```
*Premier brin*
Teneur en GC : ...
*Deuxième brin*
Teneur en GC : ...
*Nombre de mutations*
Distance de Hamming entre les deux brins : ...
```

Vérifier le bon fonctionnement de votre fonction en la testant sur les brins d'ADN `brin1` et `brin2`.

### EXTRAIRE DES DONNÉES RÉELLES D'UN FICHIER TEXTE POUR LES REPRÉSENTER

7. (a) Le fichier `hauteurs.txt` contient une seule ligne sur laquelle figurent les hauteurs des vagues, en mètres, mesurées par la balise maritime irlandaise M4 tous les jours à 11h du 15 avril 2003 au 22 novembre 2018.

Les données sont séparées par un espace ' '.

Écrire une fonction `extraction1()` sans argument qui renvoie la liste des nombres écrits dans le fichier `hauteurs.txt`.

(b) Écrire une fonction `vagues()` qui affiche un graphique représentant l'évolution de la hauteur des vagues de 2003 à 2018.

(c) Quelle est la hauteur de la plus grande vague mesurée durant cette période? .....

8. (a) Le fichier `pluiesoleil.txt` contient deux lignes.

Il s'agit d'un jeu de données généré à partir du modèle ALADIN de Météo France sur une période de référence de 1961 à 1990 pour le mois de janvier et pour l'archipel de Molène (en Bretagne).

Sur la première ligne figurent les précipitations pluvieuses journalières en *mm*.

Sur la deuxième ligne figurent le rayonnement solaire descendant correspondant en  $W.m^{-2}$ .

Sur chaque ligne, les données sont séparées par un espace ' '.

Écrire une fonction `extraction2()` sans argument qui renvoie deux listes : une contenant les précipitations et l'autre les rayonnements solaires.

(b) Écrire une fonction `pluieSoleil()` qui affiche le nuage des points représentant ce jeu de données (rayonnement solaire en fonction des précipitations pluvieuses).