
Devoir surveillé 2

Dans l'écriture de vos programmes en Python, indiquez clairement l'indentation et veillez à respecter la syntaxe du langage. Expliquez succinctement ce que font vos programmes, notamment en commentant soigneusement votre code, et n'hésitez pas à donner des noms explicites aux variables.

L'utilisation de tout matériel électronique est interdite.

Ce sujet contient quatre questions de cours, un exercice et deux problèmes indépendants.

Durée : 2h.

Applications directes du cours

1. Tracer, avec mille points, la courbe de la fonction $f: x \mapsto x + \sin(x)$ sur l'intervalle $[-7; 7]$.
2. Écrire une fonction qui prend en argument une liste L et un objet x , et qui renvoie `True` si x est un élément de la liste L , `False` sinon.
3. Écrire une fonction qui prend en argument une liste de nombres et qui la trie dans l'ordre croissant, en utilisant l'algorithme de tri par insertion.
4. Écrire une fonction qui prend en argument un entier n et qui renvoie le terme u_n de la suite définie par :

$$\begin{cases} u_0 = 5 \\ \forall k \in \mathbb{N}, u_{k+1} = \sqrt{2 + u_k}. \end{cases}$$

Exercice — Pile pile face (d'après un oral Agro-Véto 2024)

Soit $p \in [0; 1]$. On dispose d'une pièce de monnaie qui tombe sur pile avec probabilité p .

Dans la suite, on codera le côté pile par 1 et le côté face par 0.

1. (a) Écrire une fonction qui prend en argument la probabilité p et qui simule un lancer de la pièce.
(b) Écrire une fonction qui prend en argument la probabilité p et un entier N , et qui renvoie la liste des résultats obtenus sur N lancers de la pièce.
2. (a) Écrire une fonction qui prend en argument la probabilité p , qui simule des lancers de la pièce jusqu'à obtenir la configuration pile-pile-face, et qui renvoie le nombre de lancers nécessaires à l'obtention de cette configuration.
(b) À partir de la fonction précédente, écrire une fonction qui estime le temps moyen d'attente de cette configuration.

Problème 1 — The Matrix Recoded

Les deux parties de ce problème sont indépendantes.

Partie I : Opérations matricielles

1. Écrire une fonction qui prend en arguments deux matrices et qui renvoie leur somme si elle est bien définie, et un message d'erreur sinon.

2. Écrire une fonction qui prend en arguments deux matrices A et B , et qui renvoie le produit AB s'il est bien défini et un message d'erreur sinon.

Partie II : Matrices magiques (d'après un oral Agro-Véto 2025)

On dit qu'une matrice M est une **matrice magique** lorsque la somme des éléments de chaque ligne et de chaque colonne de M est la même.

Par exemple, la matrice $\begin{pmatrix} 11 & 15 & 19 \\ 21 & 7 & 17 \\ 13 & 23 & 9 \end{pmatrix}$ est magique car la somme sur chacune de ses lignes et chacune de ses colonnes vaut 45.

Mais la matrice $\begin{pmatrix} 6 & 7 & 8 \\ 8 & 7 & 6 \end{pmatrix}$ n'est pas magique car la somme des éléments de sa première colonne vaut 14 alors que celle des éléments de sa première ligne vaut 21.

3. (a) Écrire une fonction qui prend en arguments une matrice M et un indice i de ligne de M , et qui renvoie la somme des éléments de la ligne i de la matrice M .
 (b) Écrire une fonction qui prend en arguments une matrice M et un indice j de colonne de M , et qui renvoie la somme des éléments de la colonne j de la matrice M .
4. Écrire une fonction qui prend en argument une matrice et qui renvoie `True` si la matrice est magique, `False` sinon.

Problème 2 — Chiffrement de César

Jules César aimait à crypter ses correspondances militaires. Pour cela, il utilisait une méthode extrêmement élémentaire : le chiffrement par décalage, qui consiste à décaler les lettres dans l'alphabet d'un certain nombre de lettres.

30 C. SUETONI TRANQUILLI DE VITA CAESARUM

die pervenit. Epistulae quoque eius ad senatum extant 6
 quas primum videtur ad paginas et formam memorialis
 libelli convertisse, cum antea consules et duces non
 nisi transversa charta scriptas mitterent. Extant et ad
 5 Ciceronem, item ad familiares domesticis de rebus, in
 quibus, si qua occultius perferenda erant, per notas
 scripsit, id est sic structo litterarum ordine ut nullum
 verbum effici posset: quae si qui investigare et perse-
 qui velit, quartam elementorum litteram, id est D, pro
 10 A et perinde reliquas commutet. Feruntur et [†]aitvero 7

FIGURE 1 – Description du chiffrement de César par Suétone, dans *Vie des douze Césars*.

Dans ce problème, on se propose tout d'abord d'implémenter le cryptage et le décryptage d'un message selon un codage substitutif quelconque. On implémentera ensuite le chiffrement de César, puis on en proposera une méthode de déchiffrement.

On supposera que les textes à crypter sont tous composés de lettres majuscules¹ non accentuées et d'espaces, sans ponctuation.

Partie I : Substitutions monoalphabétiques

Un **chiffrement par substitution monoalphabétique** est une méthode de cryptage très simple qui consiste à remplacer chaque lettre de l'alphabet par une autre.

La permutation des lettres utilisée pour un tel chiffrement se représente comme un dictionnaire, dont les clés sont les lettres de l'alphabet et les valeurs les lettres correspondantes après substitution.

Par exemple, le dictionnaire suivant

```
Codico = {'A': 'F', 'B': 'L', 'C': 'R', 'D': 'W', 'E': 'V',  
          'F': 'P', 'G': 'D', 'H': 'O', 'I': 'I', 'J': 'Z',  
          'K': 'A', 'L': 'N', 'M': 'E', 'N': 'B', 'O': 'H',  
          'P': 'M', 'Q': 'X', 'R': 'U', 'S': 'S', 'T': 'J',  
          'U': 'K', 'V': 'C', 'W': 'Q', 'X': 'T', 'Y': 'Y',  
          'Z': 'G', ' ': ' '}
```

représente le chiffrement par substitution dans lequel la lettre "A" devient "F", la lettre "B" devient "L" et ainsi de suite.

1. Écrire une fonction **cryptage** qui prend en arguments un tel dictionnaire de codage et une chaîne de caractères, et qui renvoie la chaîne de caractères cryptée correspondante.

Exemple : l'instruction

```
| cryptage(Codico, "ALEA IACTA EST")
```

renverra la chaîne de caractères "FNVF IFRJF VSJ".

2. (a) Écrire une fonction qui prend en argument un dictionnaire de codage et qui renvoie le dictionnaire réciproque, c'est-à-dire le dictionnaire de décodage de la substitution.
(b) En déduire une fonction **decryptage** qui prend en arguments un dictionnaire de codage et un message codé (avec ce codage), et qui renvoie le message décodé.

Exemple : l'instruction

```
| decryptage(Codico, "FNVF IFRJF VSJ")
```

renverra la chaîne de caractères "ALEA IACTA EST".

Partie II : Chiffrement par décalage

3. Écrire une fonction qui prend en argument un entier d et qui renvoie le dictionnaire associé au chiffrement de César pour un décalage de d lettres (vers la droite) dans l'ordre alphabétique.

Par exemple, avec un décalage de 6 lettres, le message "VENI VIDI VICI" sera codé comme "BKTO BOJO BOIO".

1. Jules César écrivait en majuscules...

Si l'on est destinataire d'un message codé, alors on connaît très probablement le dictionnaire de codage, donc il est facile de décrypter le message (avec les fonctions de la partie précédente).

Cependant, lorsque l'on intercepte un message codé, il est rare que l'on dispose également du dictionnaire de codage. Dans le cas du chiffrement de César, pour trouver rapidement — c'est-à-dire plus rapidement qu'en testant toutes les possibilités — le nombre d de décalages effectués, on peut utiliser une analyse fréquentielle du texte à décrypter, c'est-à-dire trouver la fréquence d'apparition de chacune des lettres.

En français, la lettre la plus fréquente (dans un texte suffisamment long) est la lettre "E", la deuxième plus fréquente est la lettre "A".² Pour décoder un message, on pourra alors tester en priorité le décalage pour lequel la lettre la plus fréquente du message codé représente la lettre "E".

4. Écrire une fonction qui prend en arguments une chaîne de caractères et un caractère, et qui renvoie le nombre d'occurrences du caractère dans la chaîne.
5.
 - (a) Écrire une fonction qui prend en argument une chaîne de caractères et qui renvoie le dictionnaire dont les clés sont les caractères de la chaîne et les valeurs leur nombre d'occurrences dans la chaîne.
 - (b) En déduire une fonction qui prend en argument une chaîne de caractères et qui en renvoie le caractère le plus fréquent qui n'est pas une espace.
En cas d'ex æquo, on renverra n'importe laquelle des lettres dont le nombre d'occurrences dans la chaîne est maximal.
6. Écrire une fonction qui prend en argument un message crypté par le chiffrement de César et qui décode la chaîne en supposant que la lettre la plus fréquente dans le message d'origine est bien la lettre "E".

2. En latin, ce sont les lettres "I" et "E".