

### Exercice n°1 :

Ecrire un programme Python qui permet de :

- Saisir un nombre  $n > 2$
- Remplir aléatoirement un tableau avec  $n$  entiers de deux chiffres
- Trier ce tableau à l'aide de deux méthodes
  - **Tri par sélection**
  - **Tri à bulle**
- Afficher le contenu du tableau trié

### Exercice n°2 :

Un compte en banque est enregistré pour chaque client avec les spécifications suivantes :

Le **nom** de client (une chaîne de caractères), le **numéro de compte** (un entier), et le **solde** (un réel).

La banque désire connaître les clients qui ont un solde inférieur à une barre donnée.

**Travail demandé**

- 1- Déclarer un type enregistrement pour cette structure.
- 2- Déclarer un tableau d'enregistrements pour cet enregistrement.
- 3- Ecrire un programme nommé Banque, qui :
  - a. saisit le nombre de clients  $N$  avec ( $5 \leq N \leq 100$ ).
  - b. Range les informations de chaque client dans un tableau  $T$ .
  - c. affiche les noms, et les soldes des clients dont le montant du solde est inférieur à une barre donnée.

### Exercice n°3 :

On désire créer un fichier de données « nombres.dat » puis le remplir par des entiers. Réaliser les traitements suivants :

- 1- Remplir le fichier par des entiers. La saisie s'arrête lorsque l'utilisateur répond à la question « voulez-vous Quitter ? O/N »
- 2- Afficher le contenu du fichier.
- 3- Calculer la somme des éléments du fichier.
- 4- Vérifier l'existence d'un entier  $x$  donné.
- 5- Chercher le maximum dans le fichier.
- 6- Transférer les valeurs positives du fichier vers un autre fichier (postifs.dat).
- 7- Trier le fichier dans l'ordre croissant.

### Exercice n°4 :

Parmi les méthodes de chiffrement utilisant un mot-clé, on cite celle décrite ci-après qui permet de crypter un message ne dépassant pas 18 caractères et formé uniquement de lettres minuscules, de chiffres et d'espace :

Etape1 : Remplir aléatoirement une matrice carrée **M1** de dimension **6 \* 6** par toutes les lettres alphabétiques minuscules ainsi que tous les chiffres.

Etape 2 : Générer un message intermédiaire **msgi**, en concaténant les résultats du chiffrement de chaque caractère du message **msg**. Le résultat du chiffrement d'un caractère est la concaténation d'un caractère dont le code ASCII est égal au code ASCII de la lettre "A" auquel on ajoute l'indice de la ligne de la case contenant le caractère à chiffrer, et du caractère dont le code ASCII est égal au code ASCII de la lettre "A" auquel on ajoute l'indice de la colonne de la case contenant le caractère à chiffrer.

Le caractère espace ne sera pas chiffré.

**Etape 3 :** Remplir une deuxième matrice **M2** de taille **7\*6** caractères en mettant dans :

- La première ligne, les lettres d'un mot-clé formé de 6 lettres majuscules.
- Le reste des lignes, le message **msgi** caractère par caractère en commençant par la première case de la deuxième ligne.

**NB :** chaque case vide de la matrice **M2** sera remplie par le caractère espace

**Etape 4 :** Trier les éléments de la 1<sup>ère</sup> ligne de **M2** selon un ordre alphabétique croissant , sachant que tout déplacement d'un élément entraine le déplacement de tous les éléments de la colonne correspondante.

**Etape 5 :** concaténer les lettres de la matrice **M2**, colonne par colonne en commençant par la 1<sup>ère</sup> colonne et sans concaténer les éléments de la 1<sup>ère</sup> ligne, pour obtenir le message chiffré final.

**Exemple :**

Pour msg="promotion bac 2019" et le mot-clé "CHAISE"

**Etape 1 :** La matrice **M1** est remplie aléatoirement comme suit :

	0	1	2	3	4	5
0	c	1	o	f	w	j
1	y	m	t	5	b	4
2	i	7	a	2	8	s
3	p	3	0	q	h	x
4	k	e	u	l	6	d
5	v	r	g	z	n	9

Le résultat de chiffrement de la lettre "p" est la concaténation de la lettre "D" et la lettre "A"  
 En effet :

- Le code ASCII de la lettre "D" = code ASCII de la lettre "A" + 3 (indice de la ligne)
- Le code ASCII de la lettre "A" = code ASCII de la lettre "A" + 0 (indice de la colonne)

**Etape 2 :** le résultat du chiffrement du message msg caractère par caractère donne :

Caractère à chiffrer	p	r	o	m	o	t	i	o	n		b	a	c		2	0	1	9
Résultat du chiffrement	DA	FB	AC	BB	AC	BC	CA	AC	FE		BE	CC	AA		CD	DC	AB	FF

D'où le **msgi** est le suivant : "DAFBACBBACBCCAACFEBCCAACDDCABFF"

**Etape 3 :** Remplissage de la matrice **M2**

	0	1	2	3	4	5
0	C	H	A	I	S	E
1	D	A	F	B	A	C
2	B	B	A	C	B	C
3	C	A	A	C	F	E
4		B	E	C	C	A
5	A		C	D	D	C
6	A	B	F	F		

**Etape 4 :** Tri de la matrice **M2**

	0	1	2	3	4	5
0	A	C	E	H	I	S
1	F	D	C	A	B	A
2	A	B	C	B	C	B
3	A	C	E	A	C	F
4	E		A	B	C	C
5	C	A	C		D	D
6	F	A		B	F	

**Etape 5 :**

Le message chiffré est "FAAECFDBC AACCEAC ABAB BBCCCDFABFCD".

**On se propose d'écrire un programme permettant :**

- de saisir un message ne dépassant pas **18** caractères et formé uniquement de lettres minuscules, de chiffres et d'espaces.
- de saisir un mot-clé formé de **6 lettres majuscules**.
- de remplir aléatoirement une matrice carrée **M1** de dimension **6 \* 6** par toutes les lettres alphabétiques minuscules ainsi que tous les chiffres.
- de crypter le message **msg** selon la méthode de chiffrement décrite précédemment.
- d'afficher le message chiffré.

**Travail demandé :**

- 1) Ecrire l'algorithme du programme principal en le décomposant en modules.
- 2) Ecrire l'algorithme de chaque module envisagé.
- 3) Implémenter la solution en Python.