R1.01 Initiation au développement
Feuille de TP n°11
Manipuler des structures de données

Exercice 1 *Petit week-end entre amis - Résoudre un petit problème*

On est à la fin d'un week-end entre amis, et on cherche à faire les comptes suite aux dépenses de chacun.

Par exemple, plusieurs amis se sont retrouvés pour un week-end de mai. Ce week-end là, Pierre a acheté du pain pour 12 euros, Paul a dépensé 100 euros pour les pizzas, Pierre a payé l'essence et en a eu pour 70 euros, Marie a acheté du vin pour 15 euros, Pierre a aussi acheté du vin et en a eu pour 10 euros, Anna n'a quant à elle rien acheté.

Pierre	Paul	Marie	Anna
12	100	15	
70			
10			

A la fin du week end, les amis font les comptes pour équilibrer les dépenses. Ainsi Pierre doit recevoir 40,25 €, Paul doit recevoir 48,25 €, Marie doit verser 36,75 € et Anna doit verser 51.75 €.

Un peu plus tard au mois de juin, les amis ont organisé un nouveau week-end auquel Paul n'a pas pu participer. En revanche Béatrice et Sasha se sont jointes au groupe. Pour les comptes, Pierre a acheté du fromage pour 15 euros et du pain pour 12 euros. Marie a acheté le vin pour 20 euros et les glaces pour 34 euros. Anna a payé les pizza et en a eu pour 52 euros. Béatrice a payé 8 euros de pistaches pour l'apéro. Et Pierre a financé la location du film pour 8 euros et les pop corn pour 3 euros.

1.1 Pour le week end du mois de juin, calculer la somme que chaque participant devra verser ou recevoir pour équilibrer les dépenses.

1.2 Quelle structure de données pourrait contenir toutes les informations relatives à un week-end ? Compléter en particulier les deux exemples suivants (il est possible d'en imaginer d'autres)

```
week_end_mai = ...
week_end_juin = ...
```

1.3 Écrire le code d'une fonction `affiche_bilan_financier(week_end)` qui prend une telle structure de données en paramètre et qui affiche un bilan financier. Par exemple

```
>>> affiche_bilan_financier(week_end_mai)
Pierre doit recevoir 40,25 euros
Paul doit recevoir 48,25 euros
Marie doit verser 36,75 euros
Anna doit verser 51.75 euros
```

Votre code devra respecter toutes les bonnes pratiques vues jusqu'à présent.

- ☐ Le nom des variables et des fonctions doit être explicite
- ☐ Chaque fonction réalise une tâche clairement identifiée dans sa documentation
- ☐ Chaque fonction doit réaliser une et une seule tâche et son code fait 20 lignes maximum, sauf dans des cas exceptionnels.
- ☐ Chaque fonction doit, dans la mesure du possible, être testée.
- ☐ Le code doit être commenté si nécessaire.

Exercice 2 *Oh les amoureux !*

L'Association des Timides Maladifs a décidé de prendre les choses en main ! Chaque membre a écrit sur un petit papier son nom, et le nom de l'être aimé. Ces petits papiers ont été compilés en un dictionnaire dont les clés sont les membres de l'ATM, et la valeur associée à chaque clé est le nom de l'élu·e de son cœur.

2.1 Écrire une fonction qui à partir d'un tel dictionnnaire, renvoie les couples dont l'amour est réciproque.

2.2 Un infâme personnage s'est emparé du dictionnaire ! Il va éliminer ses rivaux et rivales ! Et vous allez l'aider… Écrire une fonction qui à partir d'un tel dictionnaire et du nom d'une personne, renvoie toutes les personnes qui sont amoureuses de la personne en question.

Exercice 3 *Encore des matrices*

Quand on a une matrice de grande taille, on est souvent amené à s'intéresser à une petite partie de la matrice. Ainsi, dans un jeu, si notre matrice représente un niveau, les *environs* du personnage sont représentés par la partie de la matrice autour du personnage. Étant donné une matrice M, la sous-matrice de format située en est la matrice à lignes et colonnes dont les coefficients sont ceux de M en partant de la case . On veut écrire une fonction qui donne la sous-matrice de taille dont le coin haut-gauche est situé à la position de . On vous donne ci-dessous quelques exemples de matrices et de sous-matrices.

Appel	Matrice	Sous-matrice
	$\begin{pmatrix}\boxed{1 & 2}& 3\\ \boxed{4 & 5}& 6\\ 7 & 8 & 9\end{pmatrix}$	$\begin{pmatrix}1 & 2\\ 4 & 5\end{pmatrix}$
	$\begin{pmatrix}1 & 2 & 3\\ 4 & \boxed{5 & 6}\\ 7 & \boxed{8 & 9}\end{pmatrix}$	$\begin{pmatrix}5 & 6\\ 8 & 9\end{pmatrix}$
	$\begin{pmatrix}1 & 2 & 3\\ \boxed{4 & 5 & 6}\\ 7 & 8 & 9\end{pmatrix}$	$\begin{pmatrix}4 & 5 & 6\end{pmatrix}$

3.1 Quelles sont les conditions sur les arguments de pour que son appel soit possible ?

3.2 Écrire la fonction

3.3 Écrire une fonction avec comme paramètres , , , qui colle dans à la position . Cette fonction est une *mutation* qui modifie la valeur de dans le tas.

Le tableau ci-dessous donne des exemples d'application de .

Position	Matrice	Sous-matrice	valeur de après l'appel
	$\begin{pmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{pmatrix}$	$\begin{pmatrix}67 & 42\\ 43 & 17\end{pmatrix}$	$\begin{pmatrix}\boxed{67 & 42}& 3\\ \boxed{43 & 17}& 6\\ 7 & 8 & 9\end{pmatrix}$
	$\begin{pmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{pmatrix}$	$\begin{pmatrix}67 & 42\\ 43 & 17\end{pmatrix}$	$\begin{pmatrix}1 & 2 & 3\\ 4 & \boxed{67 & 42}\\ 7 & \boxed{43 & 17}\end{pmatrix}$
	$\begin{pmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{pmatrix}$	$\begin{pmatrix}6 & 5 & 4\end{pmatrix}$	$\begin{pmatrix}1 & 2 & 3\\ \boxed{6 & 5 & 4}\\ 7 & 8 & 9\end{pmatrix}$

Exercice 4 *Syracuse*

À partir d'un entier n, on appelle suite de Syracuse la liste d'entiers obtenue ainsi :

- le premier entier est n
- si le dernier entier calculé k est pair, le suivant est sa moitié
- sinon, si cet entier k est impair, le suivant est $3k + 1$
- si le dernier entier calculé est 1, c'est la fin de la liste.

Voici quelques exemples :

valeur de départ	suite de Syracuse
1	1
2	2, 1
3	3, 10, 5, 16, 8, 4, 2, 1
5	5, 16, 8, 4, 2, 1

On suppose que ce processus s'arrête pour toute valeur de départ.[1]

4.1 Écrire une fonction qui renvoie la suite de Syracuse avec pour valeur de départ n (sous forme de liste d'entiers).

4.2 Combien vaut ?

On appelle *temps de vol* d'un entier n le nombre de termes avant d'arriver à 1, si on part de n.

On cherche des nombres «petits» qui aient le plus long temps de vol possible.

4.3 Écrire une fonction qui calcule le temps de vol de l'entier n.

4.4 Écrire une fonction qui renvoie le nombre inférieur à n qui a le plus long temps de vol.

4.5 Jusqu'à quelle valeur de n l'appel à prend-il moins d'une seconde ?

Voici une piste pour améliorer le temps de calcul de : on va garder dans un dictionnaire chaque temps de vol que l'on calcule ; quand on calcule un temps de vol, si on retombe sur un nombre dont on connait le temps de vol, on peut répondre directement sans avoir besoin de calculer les valeurs suivantes de Syracuse.

4.6 Sachant que le temps de vol de 10 est 6, quel est le temps de vol de 20 ? Sachant que le temps de vol de 124 est 108, quel est le temps de vol de 27 ?

4.7 Écrire une fonction qui calcule le temps de vol de n grâce aux temps de vols déjà connus, contenus dans le dictionnaire . Par exemple, pour la question précédente, avec les temps de vol que l'on connaît, on ferait un appel comme .

4.8 Modifier pour utiliser .

1. Vérifier que c'est effectivement le cas pour toute valeur de départ occupe les mathématiciens depuis 1928.