

\_\_\_\_\_ Le sujet comporte 5 exercices et 7 pages, dont une page d'annexe \_\_\_\_\_  
**Aucun document n'est autorisé** – Toutes les fonctions de manipulation d'images disponibles sont  
rappelées en annexe page 7. Vous pouvez détacher cette annexe pour plus de facilité.

---

**Exercice 1** Considérons la fonction `mystere` suivante prenant en paramètre une liste de chiffres :

```
def mystere(L):  
    s = 0  
    for i in L :  
        s = 10 * s + i  
    return s
```

1. Simulez l'exécution de l'appel `mystere([1, 2, 5, 3])` en complétant le tableau suivant montrant l'évolution des variables `s` et `i` :

<i>i</i>		
<i>s</i>		

2. Que retourne l'appel `mystere([1, 2, 5, 3])` ?

3. Que retourne l'appel `mystere([9, 6, 8])` ?

4. Comment interpréter la valeur retournée par la fonction `mystere` en général, pour une liste `L` de chiffres ?

## Exercice 2 Moyenne

On stocke les notes obtenues par les étudiants dans une liste  $L$  de réels.

1. Ecrivez une fonction `moyenneListe(L)` qui calcule et retourne la moyenne des éléments de  $L$ .

Par exemple si  $L1 = [12.0, 9.5, 10.0, 15.0, 10.5]$  `moyenneListe(L1)` retournera 11.4

A large empty rectangular box with a thin black border, intended for the student to write the implementation of the `moyenneListe(L)` function.

2. Ecrivez une fonction `nbNotesAuDessus(L)` qui calcule et retourne le nombre d'éléments de la liste  $L$  qui sont supérieurs ou égaux à la moyenne des éléments de  $L$ .

Par exemple `nbNotesAuDessus(L1)` retournera 2 car il y a seulement 2 notes supérieures ou égales à la moyenne des éléments de  $L1$  : 12.0 et 15.0.

A large empty rectangular box with a thin black border, intended for the student to write the implementation of the `nbNotesAuDessus(L)` function.

3. Ecrivez une fonction `existeNoteMoyenne(L)` qui retourne *True* s'il existe un élément de  $L$  égal à la moyenne des éléments de  $L$  et *False* sinon.  
Par exemple `existeNoteMoyenne(L1)` retournera *False* car toutes les notes de  $L1$  sont différentes de la moyenne des éléments de  $L$ .

4. Ecrivez une fonction `toutesNotesMoyenne(L)` qui retourne *True* si tous les éléments de  $L$  sont égaux à la moyenne des éléments de  $L$  et *False* sinon.

### Exercice 3 Nénuphars

On souhaite planter un nénuphar dans un bassin. Le vendeur nous prévient que les nénuphars ont une croissance très rapide et que leur diamètre double chaque semaine.

1. Nous décidons d'acheter un nénuphar de 5cm. Remplir le tableau suivant :

Nombre de semaines	0	1	2	3	4
Diamètre (en cm)	5				

2. Écrire une fonction `diametreNenuphar(d,n)` qui prend en entrée  $d$ , le diamètre actuel de notre nénuphar, ainsi qu'un nombre de semaines  $n$  et qui retourne le diamètre du nénuphar après  $n$  semaines de croissance.

3. Nous souhaitons savoir quel serait le diamètre d'un nénuphar de 8cm après 12 semaines de croissance en utilisant la fonction précédente. Comment appeler `diametreNenuphar` pour obtenir la réponse ?

4. Écrire une fonction `tientDansLeBassin(d,n,db)` qui prend en entrée  $d$ , diamètre actuel d'un nénuphar ainsi qu'un nombre de semaines  $n$  et que le diamètre du bassin  $db$ . La fonction renvoie *True* si le nénuphar ne remplit pas complètement le bassin au bout de  $n$  semaines et *False* sinon.

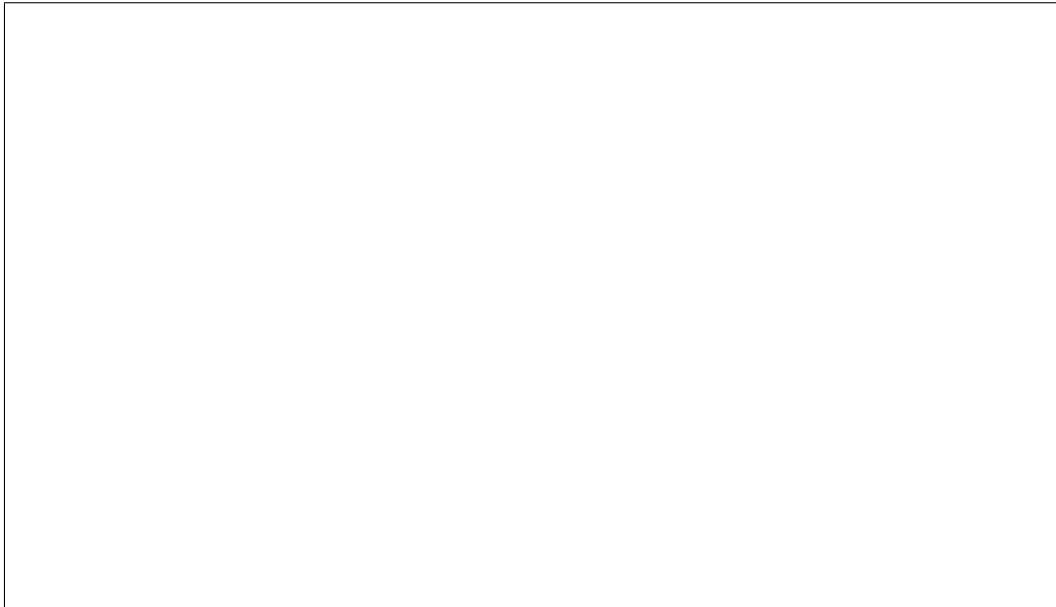
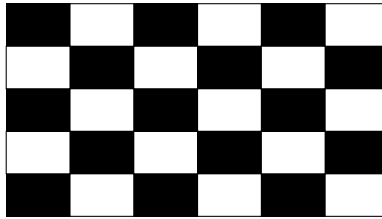
#### Exercice 4 Damier

1. Écrivez une fonction `quadrants(img)` qui dessine dans l'image `img` quatre rectangles noirs et blancs alternés. L'image résultante doit donc ressembler à ceci (sans le cadre autour qui ne sert ici qu'à vous montrer le bord de l'image) :



**2. Question à traiter en dernier !**

Écrivez une fonction `damier(img,c1,ch)`, qui dessine dans l'image `img` un damier de `c1` cases de large par `ch` cases de haut, proportionnellement aux dimensions de l'image. Un appel à `damier(img,6,5)` pourrait donc produire le résultat suivant :



### Exercice 5 Entrelacement

1. Écrivez une fonction `entrelacer(img1, img2)` qui recopie les lignes impaires de l'image `img1` dans l'image `img2`, ces deux images étant de même dimension.

2. On appelle la fonction `entrelacer` sur les images suivantes (agrandies ici 4×) :



`img1`



`img2`

Que contiendra `img2` après l'appel à cette fonction ?

*Annexe* : voici un rappel des principales fonctions disponibles pour manipuler les images (à vous de voir celles qui sont utiles pour ce devoir).

L'argument <code>img</code> est une image	
<code>open(nom)</code>	Ouvre le fichier <i>nom</i> et retourne l'image contenue dedans (par exemple <code>open("teapot.png")</code> ).
<code>Image.save(img, nom)</code>	Sauvegarde l'image <i>img</i> dans le fichier <i>nom</i> .
<code>new("RGB", (largeur, hauteur))</code>	Retourne une image de taille <i>large</i> × <i>haut</i> , initialement noire.
<code>largeur = img.width</code> <code>hauteur = img.height</code>	Récupère la largeur de <i>img</i> . Récupère la hauteur de <i>img</i> .
<code>Image.putpixel(img, (x,y), (r,g,b))</code>	Peint le pixel $(x, y)$ dans l'image <i>img</i> de la couleur $(r, g, b)$
<code>(r,g,b) = Image.getpixel(img, (x,y))</code>	Retourne la couleur du pixel $(x, y)$ dans l'image <i>img</i>