td	td 2.2 (1)	TSI1
	Images : opérations de base	Période 3
		1h

Les différentes fonctionnalités utiles des bibliothèques sont rappelées en annexe.

1 Import des bibliothèques utiles

1) Ecrire les instructions permettant d'importer les bibliothèques suivantes :

- **pyplot** sous le nom **plt**
- **imageio** sous le nom **iio**
- **numpy** sous le nom **np**

from matplotlib import pyplot as plt

import imageio as iio

import numpy as np

2 Lecture et affichage d'une image

L'image suivante a comme dimension 766 x 512 en pixels et 12,9 x 8,6 en cm.

paysage.jpg

2) Déterminer la résolution en ppp de la photographie sachant que 1po = 2,54 cm. (1po = 1 pouce)

ppp : px par po horizontalement : 766/(12,9/2,54)=151 ppp

verticalement : 512/(8,6/2,54)=151 ppp

3) Calculer la taille en mémoire vive RAM (en absence de compression) sachant que chaque couleur **RGB** est codé sur 1 octet.

RAM = 766*512*3 = 1,2 Mo

4) La taille du fichier enregistré est **73 Ko**. En déduire le taux de compression **tc** du codage **jpeg**.

tc = 1- 73/1200 = 94 %

5) Écrire une instruction permettant d'affecter à **phot1** la lecture de l'image **paysage.jpg**.

phot1 = iio.imread('paysage.png')

6) Ecrire une instruction qui permet d'afficher l'image contenu dans le tableau **phot1** de la question précédente.

plt.imshow(phot1)

plt.show()

3 Création d'une image

0 est le niveau bas d'une couleur et 255 son niveau haut.

7) Ecrire le tableau **tab0**, représentant un pixel noir entouré d'une bande blanche de largeur 1 pixel.

tab0 = [[255, 255 , 255],[255, 0, 255],[255, 255, 255]]

Pour une image **3x2** : **3** est la largeur et **2** la hauteur.

8) Définir les tableaux suivants associés à un rectangle vert de taille **3x2** :

- le tableau représentant la couleur rouge **R1 =** **[[0,0,0],[0,0,0]]**
- le tableau représentant la couleur vert **V1 =** **[[255,255,255],[255,255,255]]**
- le tableau représentant la couleur bleue **B1 =** **[[0,0,0],[0,0,0]]**
- le tableau associé à l'image du rectangle vert :

tab1 = **[[[0,255,0] , [0,255,0] , [0,255,0]] ,**
[[0,255,0] , [0,255,0] , [0,255,0]]]

9) Quelles sont les dimensions des 4 tableaux de la question précédente.

R1, V1 et B1 ont une dimension 2x3 et tab1 a une dimension 2x3x3.

10) En utilisant la bibliothèque **numpy**, écrire les instructions qui permettent de générer les tableaux **R1**, **G1** et **B1** d'une part puis d'autre part à partir de ces tableaux le tableau **tab1** du rectangle vert.

```
R1 = np.zeros((2,3),int)
V1 = 255*np.uns((2,3),int)
B1 = np.zeros((2,3),int)
tab1 = np.zeros((2,3,3),np.uint8)
tab1[:,:,1] = G1
```

11) Enregistrer ce rectangle vert sous le nom **rect.png** (il faut convertir les valeurs en entiers **uint8**).

```
iio.imsave ( 'rect.png' , np.array (tab1,uint8) )
```

12) Ecrire une fonction **luminance()** qui admet en entrée le tableau tab associé à une image RVB et qui permet d'affecter à tab1g, l'image du drapeau français en niveau de gris selon la norme :
gris = 0.299* rouge + 0.587 * vert + 0.114 * bleu. On utilisera des boucles imbriquées.

```
def luminance(tab):
    R,V,B = tab[:,:,0], tab[:,:,1], tab[:,:,2]
    nL, nC = len(R) , len(R[0])
    G = np.zeros((nL,nC),int)
    for i in range (nL) :
        for j in range (nC) :
            G[i,j] = 0.299*R[i,j] + 0.587 *V[i,j] + 0.114*B[i,j]
    return G
tab1g = luminance (tab1)
```

Annexe : Bibliothèques

Module **pyplot,** de la bibliothèque **matlplotlib,** importé sous le nom **plt**

fonctionnalité	instruction
ajouter l'image associée au tableau **tab1** à 2 ou 3 dimensions	**plt.imshow(tab1)**
Ajouter l'image en nuance de gris codé de 0 à 255 du tableau **tab1** à 2 ou 3 dimensions	**plt.imshow(tab1, cmap='gray', vmin = 0, vmax = 255)**
Montrer et fermer la figure en cours	**plt.show()**

Bibliothèque **imageio** importée sous le nom **iio**

fonctionnalité	instruction
lire un fichier **'image1.png'** enregistré dans le même dossier que le programme en cours d'édition et stocker le tableau associé dans la variable **tab1** extensions courantes : **.gif .png .jpg**	**tab1 = iio.imread('image1.png')**
Enregistrer l'image 'image1.png' associée à un tableau de valeurs tab1 extensions courantes : **.gif .png .jpg**	**iio.imsave('image1.png',tab1)**

Bibliothèque **numpy** importée sous le nom **np**

Créer un vecteur (tableau à 1 dimension) à l'aide d'une liste	**np.array([1, 2, 3])**
Créer un tableau à 2 dimensions à partir d'une liste (ici à partir d'une liste de 3 listes soient 3 lignes et 1 colonnes)	**np.array([[1],[2],[3]])**
Créer un tableau M à 2 dimensions à partir d'une liste (ici à partir d'une liste de 2 listes soient 2 lignes et 3 colonnes)	**np.array([[1,2,3],[4,5,6]])**
Créer un tableau de 0 (**float** ou **int**) à 2 lignes et 3 colonnes	**np.zeros((2,3),float)** **np.zeros((2,3),int)**
Créer un tableau de 1 (**float** ou **int**) à 2 lignes et 3 colonnes	**np.ones((2,3),float)** **np.ones((2,3),int)**

Instructions sans extension np (méthode ou opérations appliquées aux tableaux)

Accéder à un élément	**v[0], M[0,1]**
Accéder au dernier élément, et l'avant dernier	**v[-1], v[-2]**
Extraire la 2ème ligne ou 2ème colonne	**M[1,:] ou M[:,1]**
Extraire une portion de tableau (2 premières colonnes)	**M[:,0:2]**