

	td 2.2 (1)	TS11
	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 :
 - **matplotlib** sous le nom **plt**
 - **imageio** sous le nom **iio**
 - **numpy** sous le nom **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)
- 3) Calculer la taille en mémoire vive RAM (en absence de compression) sachant que chaque couleur **RGB** est codé sur 1 octet.
- 4) La taille du fichier enregistré est **73 Ko**. En déduire le taux de compression **tc** du codage **jpeg**.
- 5) Écrire une instruction permettant d'affecter à **phot1** la lecture de l'image **paysage.jpg**.

Annexe : Bibliothèques

Module **pyplot**, de la bibliothèque **matplotlib**, 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	tab1 = iio.imread('image1.png')
extensions courantes : .gif .png .jpg	
Enregistrer l'image 'image1.png' associée à un tableau de valeurs tab1	iio.imwrite('image1.png',tab1)
extensions courantes : .gif .png .jpg	

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]