

## Exercice 1

## Recherche du min/max

Soit T un tableau (une liste de liste). On donne ici un exemple :

```
1 T = [[204, 199, 204, 201, ... ],
2       [203, 198, 204, 200, ... ],
3       [...                                     ],
4       ...                                     ]
```

Pour parcourir un tableau T à 2 dimensions, il faut 2 boucles imbriquées, la 1ere pour parcourir les lignes, et la 2e pour les colonnes.

Cela donne souvent le script suivant :

*script A :*

```
1 for i in range(len(T)):
2     # i est le numero de ligne
3     for j in range(len(T[0])):
4         # toutes les lignes ont la meme longueur que T[0]
5         # j est le numero de colonne
6         ... traitement sur T[i][j]
```

*script B :*

```
1 for line in T:
2     for elem in line:
3         ... traitement sur elem
```

1. Lequel de ces scripts realise un parcours par indice ? Lequel realise un parcours par élément ?
2. Utiliser l'un de ces scripts pour écrire une fonction `min_max`, qui détermine la valeur minimale ET la valeur maximale parmi les données de la matrice.
3. Ecrire une fonction `moyenne` qui retourne la moyenne des valeurs d'une matrice.

## Exercice 2

## Compréhension de liste

Le script suivant construit une matrice M de zeros :

```
1 n = 4
2 m = 6
3 M = # a completer (1)
4 for i in range(n):
5     line = []
6     for j in range(m):
7         line.append(0)
8     # a completer (2)
```

1. Compléter le script
2. Quelle est la dimension de cette matrice ? Et quelles seraient les dimension de l'image correspondante (largeur/hauteur) ?
3. Simplifier le script après la ligne 4, en utilisant l'instruction `[0] * m`, qui est équivalent, ici, à `[0,0,0,0]`. (Ré-écrire le script)
4. Ré-écrire ce même script, en utilisant la compréhension de liste.

## Exercice 3

### Créer une image à partir d'instructions

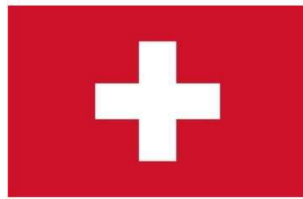


FIGURE 1 – drapeau suisse, croix blanche sur fond rouge

1. Ecrire le script qui permet de construire la matrice de ce drapeau suisse, en noir et blanc. La dimension sera de 700\*500 pixels
2. Ecrire le script qui construit la matrice du drapeau, mais cette fois, avec la couleur de fond rouge. Chaque élément de la matrice sera un tuple (R,V,B)

## Exercice 4

### Traitement d'une image sans modifier sa dimension

On souhaite transformer une image source en couleur, en une image cible en niveaux de gris. On utilise pour cela la formule de calcul de la Luminance : (*r,g,b sont les intensités des couleurs primaires 0..255*)

$$L = 0.299 \times r + 0.587 \times g + 0.114 \times b$$

La valeur calculée pour L sera placée comme niveau de gris dans la matrice cible.

1. Ecrire le script de la fonction luminance, qui calcule la valeur de L à partir d'un tuple (r, g, b)
2. Placer dans la fonction un test d'assertion sur les valeurs permises pour r,g,b. Ce test, basé sur les données d'entrée, est appelé test de *précondition*

*Rappel* : l'instruction `assert` arrête le programme lorsque la *condition* écrite dans l'instruction ne fa usse. Il s'affiche alors le *message*. L'instruction s'écrit de la manière suivante :

```
1 assert condition, "message"
```

3. Ecrire une fonction `matrice_rgb_to_gris` qui crée une matrice de niveaux de gris (valeurs de luminance) à partir d'une matrice couleur.