



IMAGE NUMÉRIQUE

Généralités et modifications

O. CHAUMETTE

Lycée Jean-Paul Sartre – 69 BRON

Informations générales concernant la séance

Nous allons travailler avec le logiciel Photofiltre pour modifier des images numériques et Python pour en créer à partir de tableaux.

***Il est conseillé d'afficher ce document à droite de l'écran et les autres logiciels à gauche.
Rappel: touche  +  ou ***

Les instructions pour manipuler les logiciels sont écrites dans des cadres bleus

Vous répondrez, sur votre cahier, aux questions des cadres marron

A la fin de la séance, vous devrez répondre à un QCM. Les réponses au QCM se trouvent parmi celles que vous aurez écrites sur votre cahier.

Préparation de la séance

téléchargez le fichier .zip des images du TP1 dans votre dossier SNT\images puis dézippez le. Il contient les images suivantes :

- ***02_Croatie.jpg***
- ***02_Maldives.jpg***

- ***02_Image_NB_1bit.py***
- ***02_Image_Gris_8bits.py***
- ***02_Image_Coul_24bits.py***

***Lancer le logiciel PHOTOFILTRE 7
Il est dans le menu démarrer \ multimedia***

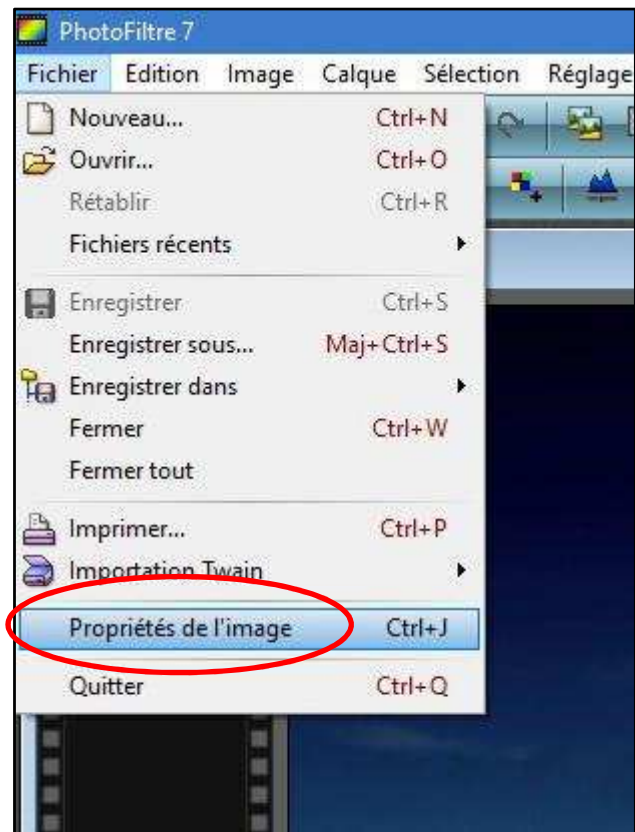
1^{ère} PARTIE

**MANIPULATION D'IMAGES
AVEC PHOTOFILTRE 7**

Caractéristiques d'une image

1. Dans Photofiltre, ouvrez l'image "02_Maldives"

2. Afficher les caractéristiques de l'image:



1. Quelles sont les dimensions de l'image en pixels?
2. Calculer le nombre total de pixels qu'elle contient en Mégapixels.
3. Quel est le poids (ou taille) du fichier image?

Nombre de couleurs d'une image

4. Combien de couleurs contient l'image ?

5. En déduire, à l'aide du tableau ci-dessous, sur combien de bits est codée l'image.

Nombre de bits	1	2	4	8	16	24
Nbre de couleurs maximal	2	4	16	256	65536	16,7 millions

Remarque: plus le nombre de bits du codage est important, plus l'image occupera de la place sur un disque dur ou une clé USB.

Accès aux réglages d'une image

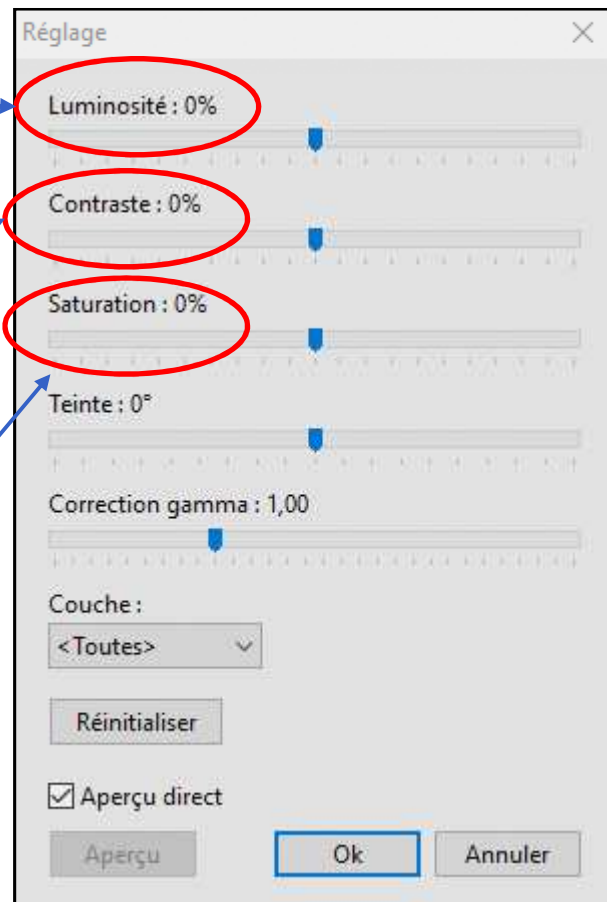
Cliquer sur les "réglages":



Luminosité: éclaircit ou assombrit chaque pixel

Contraste: augmente ou réduit l'écart entre pixels sombres et clairs

Saturation: augmente ou diminue l'intensité des couleurs



Les autres réglages ne seront pas étudiés cette année

Ajustement de la luminosité d'une image

Actionner les curseurs "Luminosité " et "contraste" pour obtenir une image plus claire et contrastée:



Pour obtenir un bon résultat, il faudra peut-être utiliser plusieurs fois de suite les réglages

Ajustement de la saturation d'une image

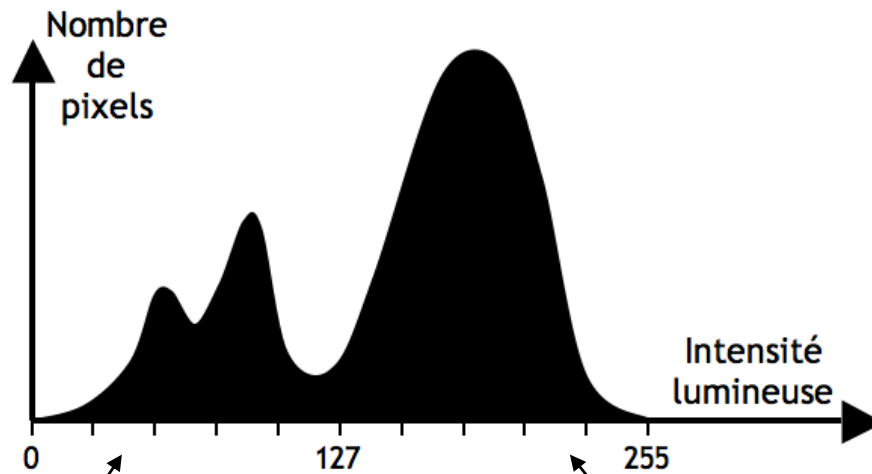
Actionner le curseur "saturation" pour obtenir une image dont les couleurs sont plus intenses:



Sauvegarder l'image sous le nom:
02_Maldives_corrected (vous l'enverrez par mail à votre professeur de SNT à la fin)

Utilisation de l'histogramme

C'est une courbe qui indique le nombre de pixels sombres et clairs de l'image

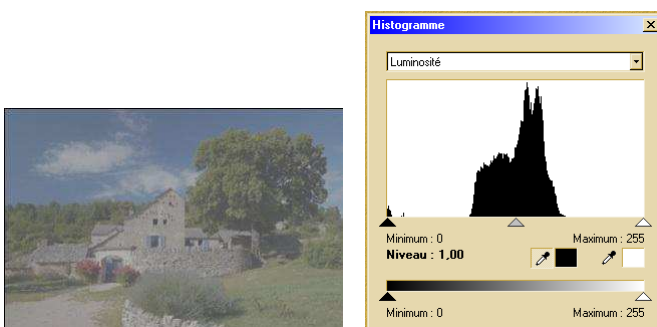


Pixels Sombres

Pixels Clairs

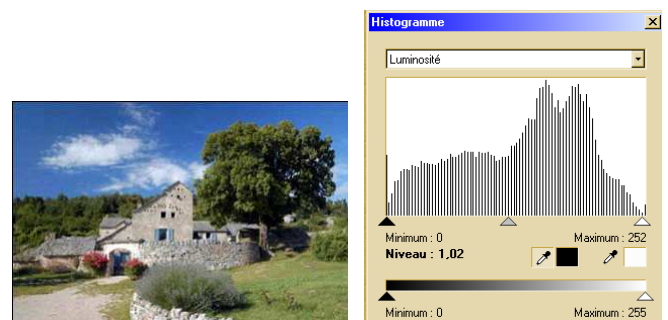
Dans cet exemple, l'image correspondante possède beaucoup de pixels clairs (grosse « bosse » à droite) et peu de pixels sombres (« bosse » plus faible à gauche).

Un mauvais histogramme:



Il n'y a pas de pixels sombres ni de pixels clairs. Beaucoup sont gris.

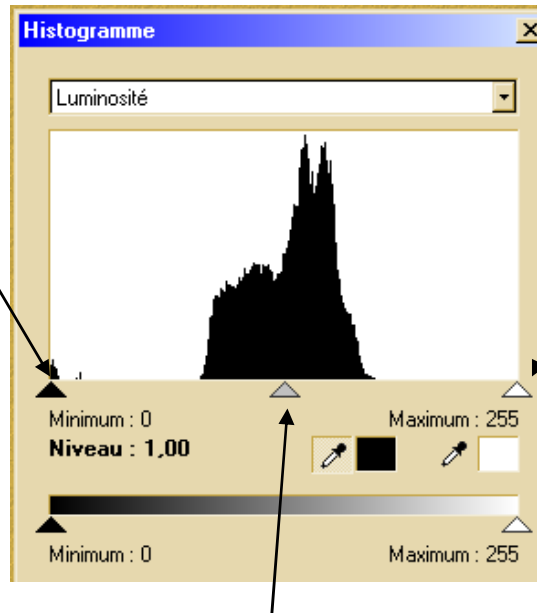
Un bon Histogramme:



Il y a des pixels sombres et clairs bien répartis

Comment agir sur l'histogramme ?

Pour assombrir:
Déplacer ce
 curseur
 vers la droite



Pour éclaircir:
Déplacer ce
 curseur
 vers la gauche

Pour diminuer la différence entre clair et foncé:
Déplacer ce curseur vers la droite

EXERCICE

1. Ouvrir l'image "02_Croatie"
2. Afficher l'histogramme
3. Modifier l'histogramme pour obtenir une image de bonne qualité



Sauvegarder l'image sous le nom:

02_Croatie_corrected puis envoyez vos 2 photos corrected par mail à votre professeur de SNT



2nde PARTIE

CRÉATION ET MODIFICATION D'IMAGES AVEC PYTHON

Préparation du travail

1. Fermer Photofiltre 7

2. Ouvrir EduPython

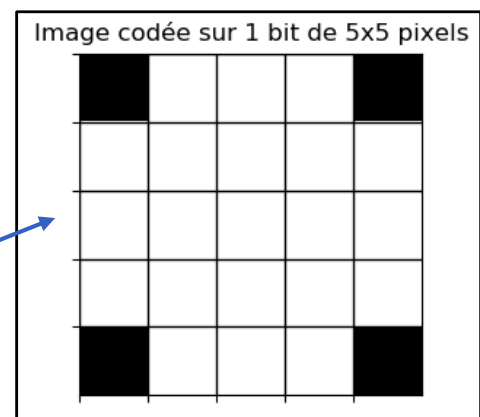
3. Charger dans l'éditeur le programme "02_Image_NB_1bit.py"

4. Lancer le programme: il affiche pour l'instant une image de 5x5 pixels en noir et blanc correspondant au tableau situé entre les lignes roses du code.

```

1 # 2 nde SNT - Simulation de la création d'une image codée sur 1 bits
# -----
# Olivier CHAUMETTE - Lycée JP Sartre - 69500 BRON
# *****
# import des modules utiles
• import numpy as np
• import matplotlib.pyplot as plt
• import matplotlib.image as img
10
• """ *****
• PARTIE ELEVES
• ***** """
# on fixe la largeur et la hauteur de l'image que l'on veut créer
# nombre de pixels suivant l'axe horizontal X
• nombre_colonnes=5
# nombre de pixels suivant l'axe vertical Y
• nombre_lignes=5
# Voici le tableau représentant l'image (rappel 0: NOIR et 1: BLANC)
• mon_image=np.array([[0,1,1,1,0],
• [1,1,1,1,1],
• [1,1,1,1,1],
• [1,1,1,1,1],
• [1,1,1,1,1],
• [0,1,1,1,0],
• ])
• """
• *****
• FIN PARTIE ELEVES
• *****
# faire un damier découverte de rajouter un 1 ou un 0
# puis 8x8 et faire un smiley
40 # on créé une image "remolie" de 0 (de taille largeur*hauteur*3 car 3 couches
02_Image_NB_1bit.py x

```



Fermer la fenêtre de l'image pour revenir au code

Création d'un Smiley de 5x5 pixels

```

PARTIE ELEVES
#####

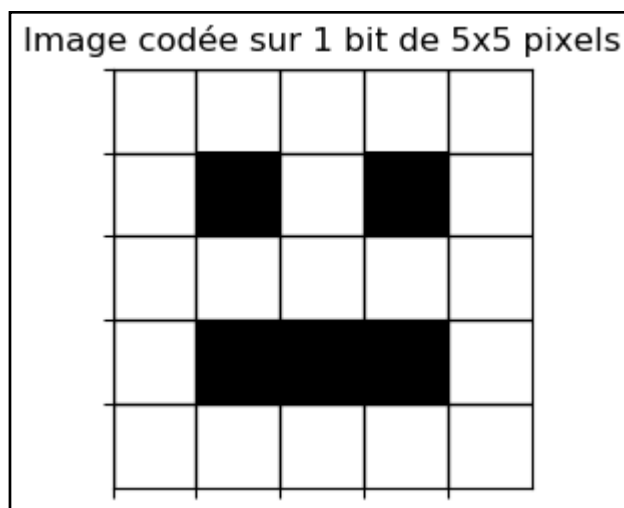
# on fixe la largeur et la hauteur de l'image que l'on veut créer
# nombre de pixels suivant l'axe horizontal X
nombre_colonnes =5
# nombre de pixels suivant l'axe vertical Y
nombre_lignes=5

# Voici le tableau représentant l'image (rappel 0: NOIR et 1: BLANC)
mon_Image=np.array([[0,1,1,1,0],
                    [1,1,1,1,1],
                    [1,1,1,1,1],
                    [1,1,1,1,1],
                    [0,1,1,1,0],
                    ])

"""
#####
FIN PARTIE ELEVES

```

Modifier le tableau dans le code pour obtenir l'image ci-dessous:



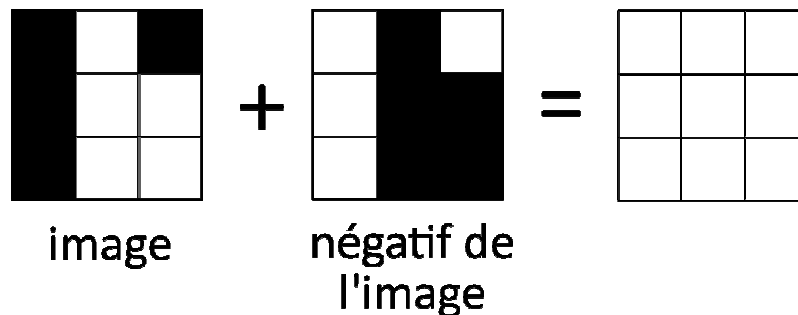
APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE IMAGE



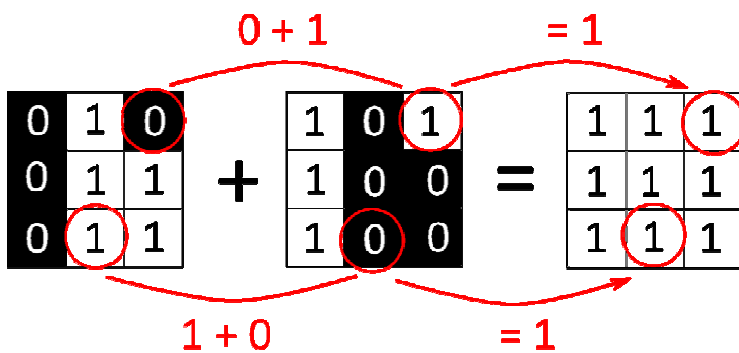
Obtention de l'image négative du Smiley

COURS

Quand on superpose une image et son "négatif", on obtient une image blanche.



Explication pour ceux qui veulent aller plus loin...



Pour chaque pixel, Si on additionne la valeur du pixel de l'image et celle du même pixel du négatif, on obtient la valeur du blanc (1 en codage 1 bit)

TRAVAIL À FAIRE

Modifier à nouveau le tableau pour obtenir l'image "négative" du smiley



APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE IMAGE

Création complète d'une image 1 bit

```

PARTIE ELEVES
#####

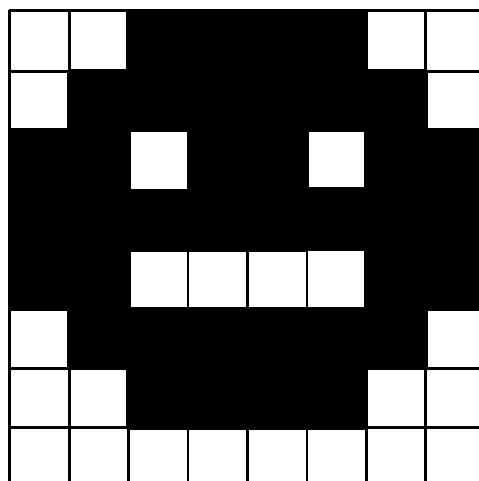
# on fixe la largeur et la hauteur de l'image que l'on veut créer
# nombre de pixels suivant l'axe horizontal X
nombre_colonnes =5
# nombre de pixels suivant l'axe vertical Y
nombre_lignes=5

# Voici le tableau représentant l'image (rappel 0: NOIR et 1: BLANC)
mon_Image=np.array([[0,1,1,1,0],
                    [1,1,1,1,1],
                    [1,1,1,1,1],
                    [1,1,1,1,1],
                    [0,1,1,1,0],
                    ])

"""
#####
FIN PARTIE ELEVES

```

Modifier et compléter le tableau ainsi que les dimensions de l'image pour obtenir:



APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE IMAGE



Création d'une image 8bits en niveau de gris

1. Charger dans l'éditeur le programme "02_Image_Gris_8bits.py"

4. Lancer le programme: il affiche pour l'instant une image de 4x4 pixels en niveau de gris correspondant au tableau situé entre les lignes roses du code.

```

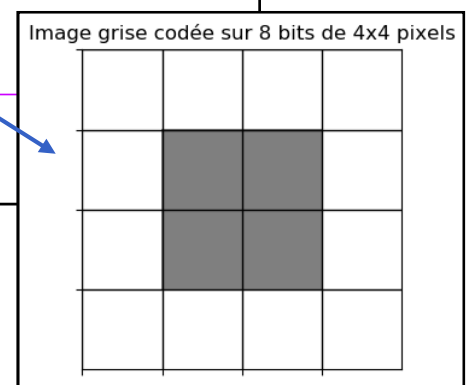
""" #####
    PARTIE ELEVES
    #####

# on fixe la largeur et la hauteur de l'image que l'on veut créer
# nombre de pixels suivant l'axe horizontal X
nombre_colonnes =4
# nombre de pixels suivant l'axe vertical Y
nombre_lignes=4

# Voici le tableau représentant l'image (rappel 0: NOIR et 255: BLANC)
mon_Image=np.array([[ 255,255,255,255],
                    [ 255,127,127,255],
                    [ 255,127,127,255],
                    [ 255,255,255,255],
                    ])

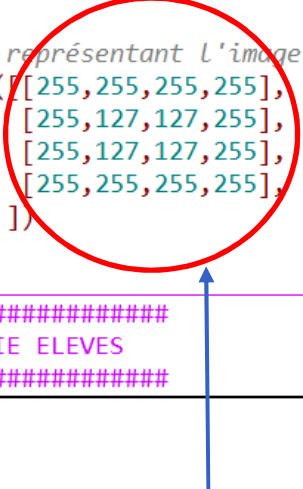
"""
#####
    FIN PARTIE ELEVES
    #####

```



Assombrissement d'une image 8 bits

```
""" #####  
    PARTIE ELEVES  
    #####  
  
# on fixe la largeur et la hauteur de l'image que l'on veut créer  
# nombre de pixels suivant l'axe horizontal X  
nombre_colonnes =4  
# nombre de pixels suivant l'axe vertical Y  
nombre_lignes=4  
  
# Voici le tableau représentant l'image (rappel 0: NOIR et 255: BLANC)  
mon_Image=np.array([[255,255,255,255],  
                    [255,127,127,255],  
                    [255,127,127,255],  
                    [255,255,255,255],  
                    ])  
  
"""  
#####  
    FIN PARTIE ELEVES  
    #####
```



Modifier le tableau dans le code pour assombrir tous les pixels d'une valeur 100

APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE IMAGE



Eclaircissement d'une image 8bits

Modifier le tableau dans le code pour éclaircir tous les pixels d'une valeur 200

***L'image présente-t-elle encore du contraste ?
(c'est-à-dire un différence importante entre pixels
sombres et clairs)***

**APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE IMAGE
ET LUI DONNER VOTRE RÉPONSE À LA QUESTION**



Modification d'une image 8bits en niveau de gris

```

""" #####
    PARTIE ELEVES
    #####

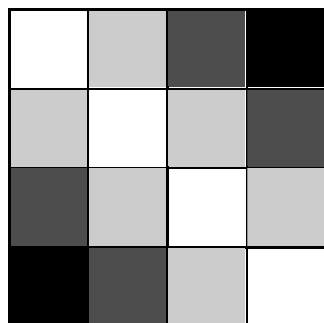
# on fixe la largeur et la hauteur de l'image que l'on veut créer
# nombre de pixels suivant l'axe horizontal X
nombre_colonnes =4
# nombre de pixels suivant l'axe vertical Y
nombre_lignes=4

# Voici le tableau représentant l'image (rappel 0: NOIR et 255: BLANC)
mon_Image=np.array([ [255,255,255,255],
                    [255,127,127,255],
                    [255,127,127,255],
                    [255,255,255,255],
                    ])

"""
#####
    FIN PARTIE ELEVES
    #####

```

Modifier le tableau dans le code pour obtenir l'image ci-dessous:



C'est à vous de choisir les "niveaux de gris" pour que votre image réponde à la consigne.

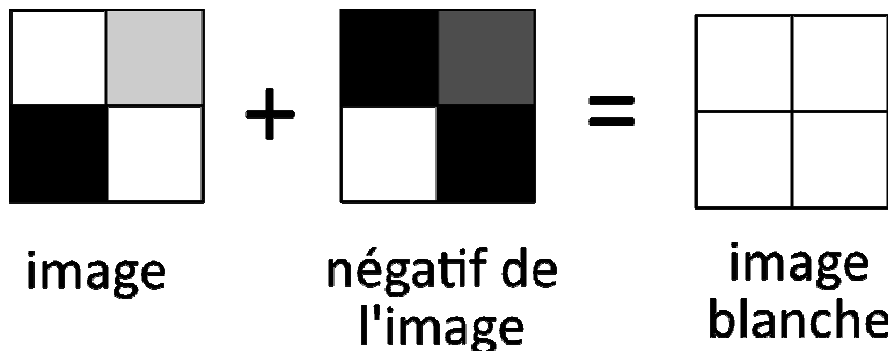


APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE IMAGE

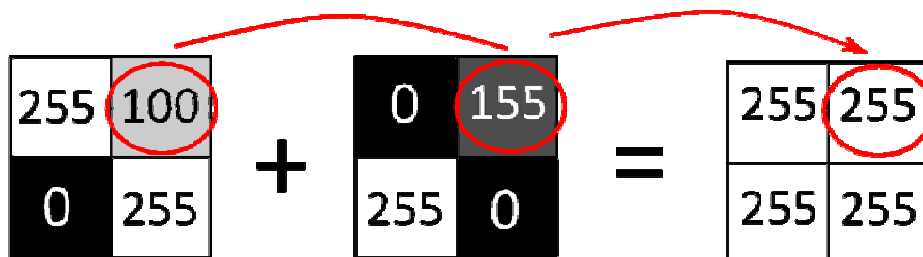
Négatif d'une image 8 bits en niveau de gris

Rappel

Quand on superpose une image et son "négatif", on obtient une image blanche.



quand on additionne la valeur d'un pixel de l'image (exple: 100) et celle du même pixel de l'image négative (exple: 155) on retrouve la valeur du blanc: 255

**TRAVAIL À FAIRE**

Modifier à nouveau le tableau pour obtenir l'image "négative" de la précédente



APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE IMAGE

Automatisation de l'obtention du négatif

Dans le code, aller au niveau de la boucle qui parcourt tous les pixels de l'image pour récupérer leur valeur

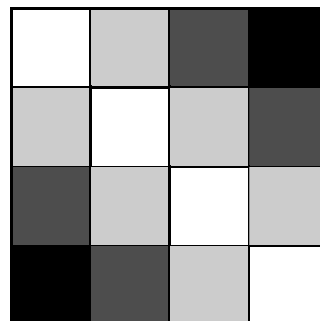
```
for y in range(0,hauteur):
    for x in range (0,largeur):
        # on mets dans la variable "ancienne_valeur_pixel"
        # |la valeur du pixel en y,x de l'image
        ancienne_valeur_pixel=mon_Image[y][x]

        """ PARTIE MODIFIABLE PAR LES ELEVES"""
        # taper ici le code pour donner au pixel une nouvelle valeur
        nouvelle_valeur_pixel=ancienne_valeur_pixel

        """ FIN PARTIE MODIFIABLE PAR LES ELEVES """
```

Modifier la ligne pour que la nouvelle valeur du pixel soit le "négatif" de l'ancien

Vous devriez obtenir:



APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE CODE

Création d'une image couleur 24 bits

1. Charger dans l'éditeur le programme "02_Image_Coul_24bits.py"

4. Lancer le programme: il affiche pour l'instant une image de 3x3 pixels en couleur correspondant au tableau situé entre les lignes roses du code.

```

""" #####
    PARTIE ELEVES
    ##### """

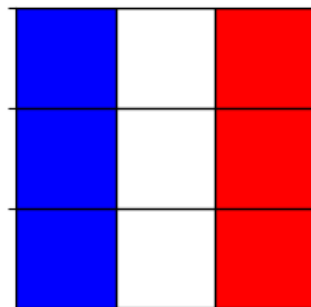
# on fixe la largeur et la hauteur de l'image que l'on veut créer
# nombre de pixels suivant l'axe horizontal X
nombre_colonnes =3
# nombre de pixels suivant l'axe vertical Y
nombre_lignes=3

# Voici le tableau représentant l'image
# rappel pour chaque pixel: [Rouge,Vert,Bleu] et 0:minimum 255:maximum
mon_Image=np.array([[ [0,0,255] , [255,255,255] , [255,0,0] ],
                    [ [0,0,255] , [255,255,255] , [255,0,0] ],
                    [ [0,0,255] , [255,255,255] , [255,0,0] ],
                    ])

""" #####
    FIN PARTIE ELEVES
    ##### """

```

Image codée sur 24 bits de 3x3 pixels



Modification d'une image couleur 24 bits

```

""" #####
    PARTIE ELEVES
    ##### """

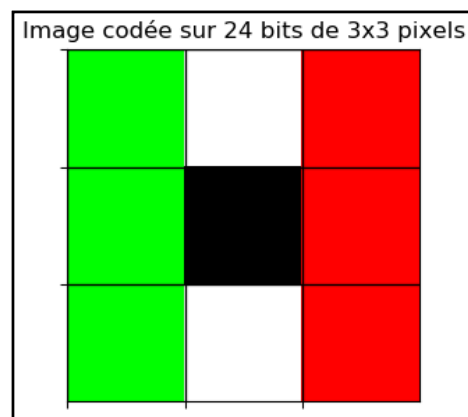
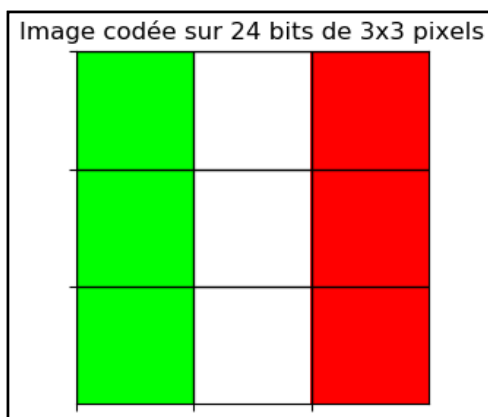
# on fixe la largeur et la hauteur de l'image que l'on veut créer
# nombre de pixels suivant l'axe horizontal X
nombre_colonnes =3
# nombre de pixels suivant l'axe vertical Y
nombre_lignes=3

# Voici le tableau représentant l'image
# rappel pour chaque pixel: [Rouge,Vert,Bleu] et 0:minimum 255:maximum
mon_Image=np.array([[ [0,0,255] , [255,255,255] , [255,0,0] ],
                    [ [0,0,255] , [255,255,255] , [255,0,0] ],
                    [ [0,0,255] , [255,255,255] , [255,0,0] ],
                    ])

""" #####
    FIN PARTIE ELEVES
    ##### """

```

Modifier le tableau dans le code pour obtenir le drapeau italien puis le drapeau mexicain:



APPELER LE PROFESSEUR POUR LUI MONTRER VOS IMAGES

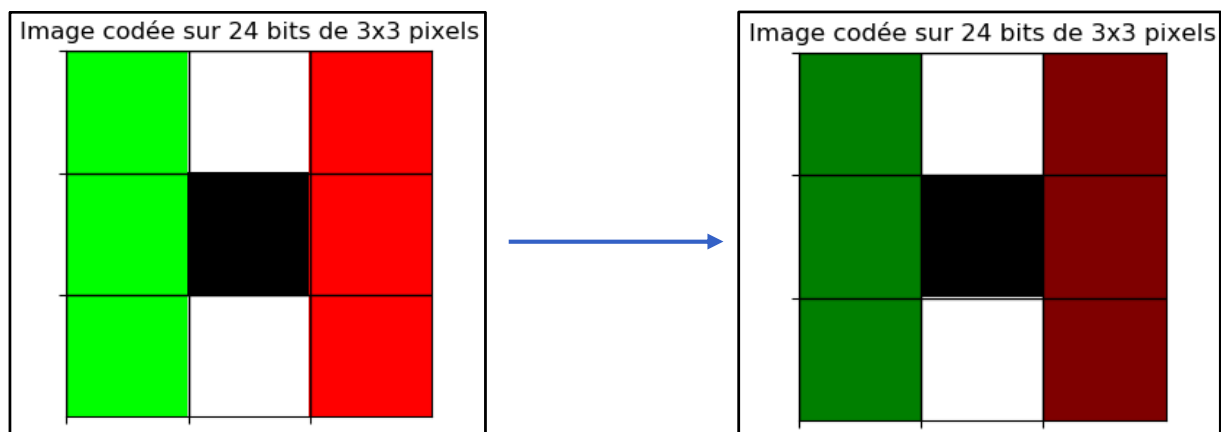
Désaturation d'une image couleur 24 bits

Rappel

Diminuer la saturation revient à diminuer l'intensité des couleurs. Le noir n'est pas touché par cette modification.

(Nous avons utilisé le réglage de la saturation avec Photofiltre)

Modifier le tableau pour que les rouges et le verts du drapeau mexicain soient moins saturés:

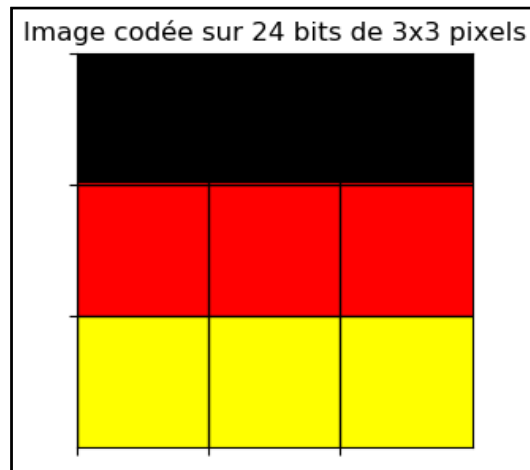


APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE IMAGE



Création d'une image couleur 24 bits

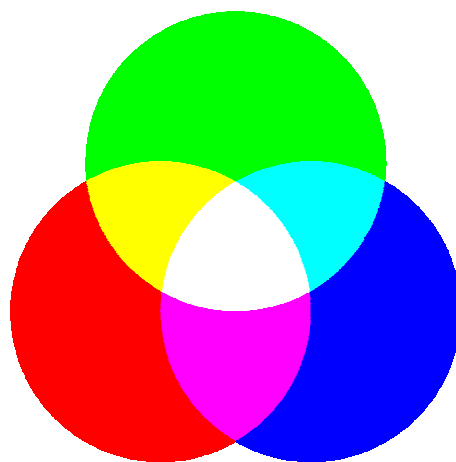
Modifier le tableau pour créer le drapeau allemand



APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE IMAGE

Rappel

Synthèse additive des couleurs:



Négatif d'une image couleur 24 bits

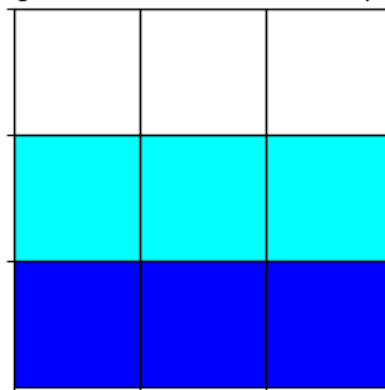
Rappel et Aide

Nous avons vu que pour obtenir le négatif d'une image, il faut soustraire la valeur du pixel à 255. En codage RVB, il faut faire cela pour chaque couleur (que l'on appelle couche).

Modifier le tableau pour créer le négatif du drapeau allemand

Vous devriez obtenir:

Image codée sur 24 bits de 3x3 pixels



APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE IMAGE

Automatisation de l'obtention du négatif

Dans le code, aller au niveau de la boucle qui parcourt tous les pixels de l'image pour récupérer leurs valeurs de rouge, vert et bleu

```

for y in range(0,hauteur):
    for x in range (0,largeur):
        # on mets dans les variables "ancien_pixel_couleur"
        # la valeur du pixel en y,x de l'image pour la couche "couleur"
        ancien_pixel_rouge =mon_Image[y][x][0]
        ancien_pixel_vert=mon_Image[y][x][1]
        ancien_pixel_bleu =mon_Image[y][x][2]

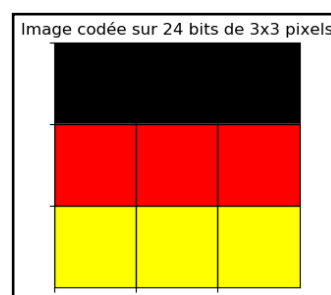
        """ PARTIE MODIFIABLE PAR LES ELEVES """
        # taper ici le code pour donner au pixel des nouvelles valeurs
        new_pixel_rouge = ancien_pixel_rouge
        new_pixel_vert = ancien_pixel_vert
        new_pixel_bleu = ancien_pixel_bleu

        """ FIN PARTIE MODIFIABLE PAR LES ELEVES """

```

Modifier les lignes pour que les nouvelles valeurs du pixel soient les "négatifs" de l'ancien

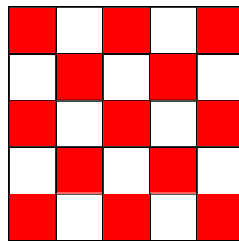
Vous devriez obtenir:



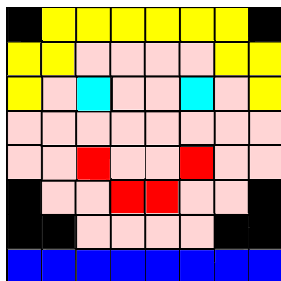
APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE CODE

POUR LES PLUS RAPIDES

Modifier le tableau et les dimensions de l'image pour dessiner le damier croate suivant:



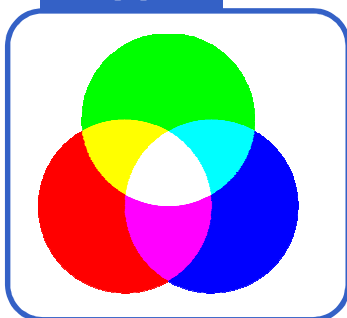
Modifier le tableau et les dimensions de l'image pour dessiner le personnage suivant:



Pour le rose, on cherchera sur internet son code RVB

<https://www.toutes-les-couleurs.com/code-couleur-rvb.php>

Rappel



**APPELER LE PROFESSEUR POUR LUI
MONTRER VOS IMAGES**

