

Le but de ce TP est de comprendre comment un ordinateur stocke les images.

Faire un compte-rendu de ce TP sur une feuille simple.

- Répondre aux questions précédées du symbole  (crayon) sur votre compte-rendu.
- Répondre sur le sujet aux questions qui ne sont pas précédées de ce symbole.
- Le symbole  (loupe) indique qu'il faut chercher la réponse à cette question sur internet.

## 0 Introduction

Copier l'ensemble du dossier partagé TP3 sur votre bureau.

## 1 Images matricielles et vectorielles

### 1.1 Première approche

1. Ouvrir les fichiers `smiley.png` et `smiley.svg`.
2.  Zoomer 2000 fois sur le bord de l'œil de `smiley.svg`. Quelle est alors la forme de cet œil ?
3.  Même question avec `smiley.png`.
4.  Comment expliquer la différence ?

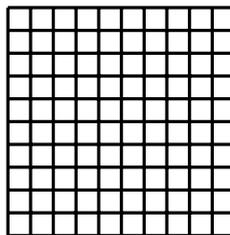
### 1.2 Différentes représentations

1. En suivant les instructions, dessiner les deux images suivantes dans les cadres de droite.
  - (a) Description *matricielle* : colorier le tableau de droite en respectant les instructions de gauche, avec les couleurs Blanc, Noir, Vert, Rouge.

```

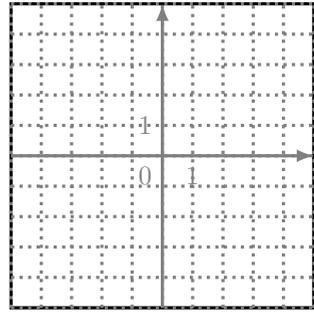
BBNNNNNNBB
BNBBBBBBNB
NBBBBBBBBN
NBBVBBVBBN
NBBBBBBBBN
NBBBBBBBBN
NBBBBBBBBN
NBBBBBBBBN
NBBRRRRBBN
BNBBBBBBNB
BBNNNNNNBB

```



- (b) Description *vectorielle* : dessiner dans le cadre de droite en suivant les instructions de gauche.

- Cercle noir de centre  $(0, 0)$  et de rayon 5.
- Disque vert de centre  $(-2, 2)$  et de rayon 0,5.
- Disque vert de centre  $(2, 2)$  et de rayon 0,5.
- Segment rouge d'épaisseur 0,5, et d'extrémités  $(-2; -2)$  et  $(2; -2)$ .

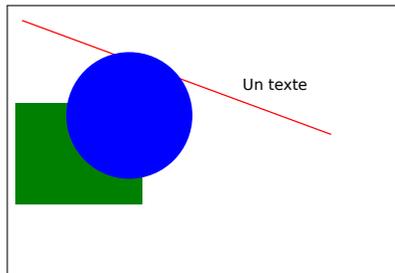


2. ✎ Laquelle des deux méthodes donne l'image la plus nette ?

### 1.3 Format SVG

Voici le code `svg` produisant l'image ci-dessous (aussi disponible dans le fichier `exemple.svg`).

```
<?xml version="1.0" encoding="utf-8"?>
<svg xmlns="http://www.w3.org/2000/svg" version="1.1" width="
  300" height="200">
  <rect width="100" height="80" x="0" y="70" fill="green" />
  <line x1="5" y1="5" x2="250" y2="95" stroke="red" />
  <circle cx="90" cy="80" r="50" fill="blue" />
  <text x="180" y="60">Un texte</text>
</svg>
```



1. Associer les formes de la figure (disque, ligne, etc.) aux lignes du fichier qui permettent de les dessiner.
2. Quelle est la signification des paramètres ?

### 1.4 Bilan

Q Une image numérique peut être représentée (principalement) de deux manières : **image matricielle** où l'image est représentée par un tableau (matrice) de couleurs. Chaque « case » du tableau est appelée \_\_\_\_\_, qui signifie \_\_\_\_\_. Les principaux formats d'image matricielle sont : \_\_\_\_\_.

**image vectorielle** où l'image est décrite par des formes géométriques. Le principal format d'image matricielle est : \_\_\_\_\_.

Aucune des deux méthodes n'est meilleure que l'autre : cela dépend de l'usage.

- Pour des dessins faits à l'ordinateur (comme des logos), une image *matricielle/vectorielle* est souvent plus adaptée.
- Pour des photos, une image *matricielle/vectorielle* est souvent plus adaptée.

## 2 Compression avec ou sans perte

1. Ouvrir le fichier `gere.jpg` avec le logiciel *Gimp*.
2. Exporter ce fichier (**Fichier > Exporter comme**) six fois, en faisant varier le nom du fichier et la qualité ou le niveau de compression :

Format JPG		Format PNG	
Nom	Qualité	Nom	Niveau de compression
<code>gere0.jpg</code>	0	<code>gere0.png</code>	0
<code>gere50.jpg</code>	50	<code>gere5.png</code>	5
<code>gere100.jpg</code>	100	<code>gere9.png</code>	9

3. Compléter le tableau suivant.

Fichier	Format (jpg/png)	Taille (en ko)	Qualité (bon/mauvais)
<code>gere0.jpg</code>			
<code>gere50.jpg</code>			
<code>gere100.jpg</code>			
<code>gere0.png</code>			
<code>gere5.png</code>			
<code>gere9.png</code>			

4. **Bilan**

**jpg** Le format d'image `jpg` utilise de la compression *avec/sans* perte : une plus grande compression se fait au détriment \_\_\_\_\_.

**png** Le format d'image `png` utilise de la compression *avec/sans* perte : une plus grande compression se fait au détriment \_\_\_\_\_.

5. **Q** Classer les formats d'image suivants suivant qu'ils utilisent une compressions avec ou sans perte de qualité : `png`, `jpg`, `tiff`, `bmp`, `gif`.

**Avec perte :**

**Sans perte :**

6. **Q** Même question avec les formats audio suivant : `mp3`, `ogg`, `flac`, `wav`.

**Avec perte :**

**Sans perte :**

### 3 Codage des couleurs

Dans cette partie, nous considérons une image matricielle (telle que définie à la partie 1.4, et nous nous intéresserons à manière dont sont codées les couleurs.

#### 3.1 Souvenirs d'art plastique

1. 🖌️ Quelles sont les trois couleurs primaires en peinture ?
2. 🖌️ Comment obtenir les couleurs suivantes à partir des trois couleurs primaires ?

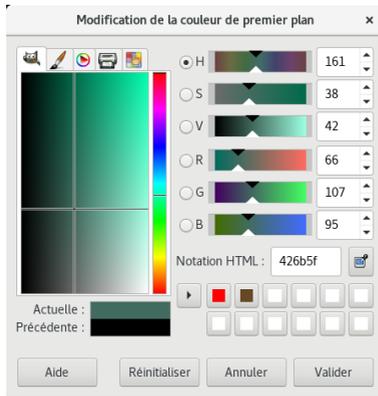
Orange :

Vert :

Violet :

#### 3.2 Modèle RGB

1. (a) 🔍 🖌️ Que signifient les initiales RGB ?  
(b) 🔍 🖌️ Pourquoi avoir choisi ces trois couleurs plutôt que les trois couleurs primaires en peinture ?
2. (a) Dans le logiciel GIMP, créer une nouvelle image (Fichier > Nouvelle image).  
(b) Cliquer sur la couleur de premier plan (le carré noir dans le menu de gauche) pour afficher la palette des couleurs.



} Ignorer pour le moment.

← Quantité de rouge

← Quantité de vert

← Quantité de bleu

- (c) 🖌️ Quelles sont les valeurs minimales et maximales que peuvent prendre les quantité de rouge, vert, bleu ?
3. Compléter le tableau suivant en utilisant la palette de Gimp utilisée à la question précédente. On notera  $(x, y, z)$  la couleur composée de  $x$  rouge,  $y$  vert et  $z$  bleu.

Couleur	Code
Rouge	( , , )
Vert	( , , )
Bleu	( , , )
_____	(0, 255, 255)
_____	(255, 0, 255)
_____	(255, 255, 0)
_____	(255, 128, 0)
Noir	( , , )
Blanc	( , , )

4.  Quels autres modèles de couleur sont aussi utilisés en informatique ?