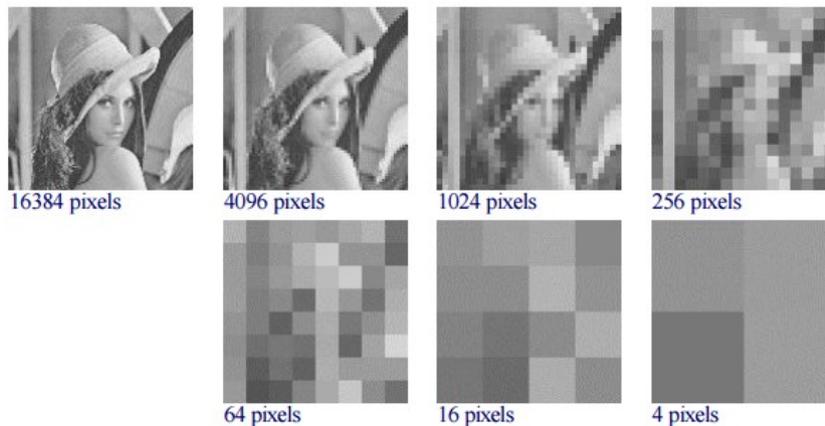


## IMAGE NUMÉRIQUE

Adapté de documents de l'université de Rennes et des académies de de Lyon et de Poitiers.

Une image numérique est divisée en pixels (picture element). Le nombre de pixels est la taille de l'image. Ci-dessous, variation du nombre de pixels pour une même image (on a gardé la même dimension, mais à l'écran, elle est divisée par 4 à chaque étape) :



**Résolution** : nombre de pixels par unité de longueur (en points par pouce, ppp ou dpi - un pouce vaut 2,54 cm). Plus la résolution est importante, plus la quantité d'informations décrivant l'objet est importante. Ci-dessus, la résolution change (il y a de moins en moins de pixels par unité de longueur). Si elle ne change pas, l'aire des images est divisée par 4 à chaque étape. Multiplier la résolution par 2 revient à multiplier le nombre de pixels de l'image par 4.

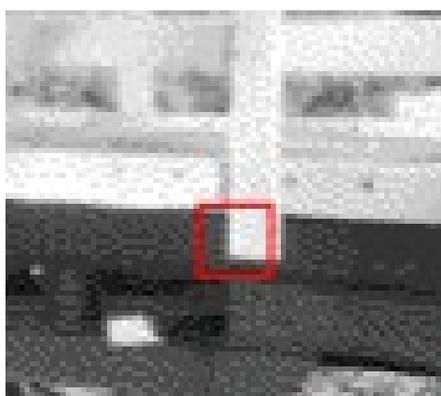
**Niveau de gris** : entre le blanc et le noir, un pixel ne peut pas prendre toutes les nuances, mais l'intensité lumineuse varie de 0 (noir) à 255 (blanc), il y a donc 256 niveaux ( $2^8$  niveaux, codage sur 8 bits). Ci-dessous : 0 à gauche, 255 à droite.



**Couleurs** : la synthèse additive permet, à l'aide de sources rouge, verte et bleue, de générer toutes les couleurs, c'est le codage RVB. Chaque teinte est généralement codée sur 8 bits.

| Rouge | Vert | Bleu | Couleur        |
|-------|------|------|----------------|
| 0     | 0    | 0    | Noir           |
| 0     | 0    | 1    | Nuance de noir |
| 255   | 0    | 0    | Rouge          |
| 0     | 255  | 0    | Vert           |
| 0     | 0    | 255  | Bleu           |
| 128   | 128  | 128  | Gris           |
| 255   | 255  | 255  | Blanc          |

**L'image numérique** : elle est donc constituée d'un tableau d'entiers comme l'exemple simple donné ci-dessous, pour une image en nuances de gris. La zone dans le petit carré au centre de l'image est en fait le tableau de nombres recopié à côté.



|     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 251 | 251 | 247 | 243 | 215 | 170 | 227 | 239 | 235 | 235 | 235 | 231 | 231 | 227 | 239 |
| 121 | 130 | 142 | 158 | 154 | 134 | 198 | 239 | 235 | 235 | 235 | 231 | 231 | 231 | 231 |
| 77  | 73  | 73  | 77  | 73  | 81  | 215 | 239 | 235 | 239 | 235 | 231 | 227 | 227 | 239 |
| 81  | 81  | 81  | 81  | 81  | 89  | 182 | 239 | 239 | 239 | 235 | 235 | 231 | 227 | 231 |
| 73  | 73  | 77  | 81  | 81  | 85  | 219 | 239 | 239 | 235 | 231 | 231 | 231 | 227 | 239 |
| 73  | 73  | 73  | 77  | 77  | 89  | 186 | 243 | 239 | 239 | 235 | 235 | 231 | 227 | 231 |
| 69  | 69  | 73  | 77  | 77  | 89  | 223 | 243 | 239 | 235 | 231 | 231 | 231 | 227 | 239 |
| 69  | 69  | 73  | 69  | 77  | 89  | 194 | 243 | 239 | 239 | 235 | 231 | 231 | 227 | 235 |
| 69  | 69  | 73  | 73  | 77  | 89  | 223 | 243 | 239 | 235 | 231 | 231 | 231 | 231 | 239 |
| 69  | 69  | 73  | 73  | 77  | 89  | 194 | 243 | 239 | 239 | 235 | 231 | 231 | 231 | 239 |
| 69  | 69  | 69  | 73  | 77  | 81  | 150 | 178 | 194 | 210 | 223 | 227 | 227 | 231 | 243 |
| 69  | 65  | 69  | 73  | 73  | 77  | 81  | 81  | 85  | 85  | 89  | 93  | 101 | 105 | 117 |
| 57  | 57  | 61  | 65  | 65  | 69  | 69  | 69  | 69  | 69  | 73  | 73  | 77  | 77  | 77  |
| 49  | 49  | 49  | 49  | 53  | 53  | 53  | 53  | 57  | 57  | 57  | 57  | 61  | 61  | 61  |
| 49  | 53  | 53  | 57  | 61  | 57  | 57  | 53  | 49  | 49  | 49  | 49  | 49  | 49  | 49  |

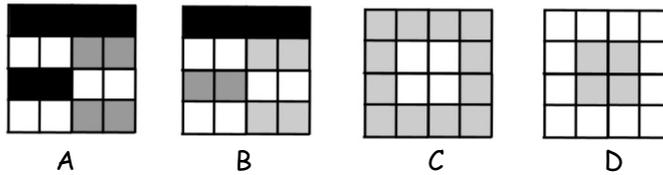
Voici ce à quoi ressemble le code binaire de ce carré, où chaque nombre est exprimé en binaire sur 8 bits.

```
1100111111001101011001010011
1011111010111001101010110101
1100010010011101010101110101
1100111010001010010000110011
```

**Remarque :** les logiciels utilisent parfois la base 16, le code hexadécimal, de sorte que les valeurs peuvent aller de 0 à F - en fait, de 00 à FF (pour 255).

**Questions.**

- 1) Combien de nuances de gris peuvent prendre les pixels d'une image si l'on code sur 8 bits les niveaux de gris ?
- 2) Ci-dessous, quatre images et des tableaux aux valeurs exprimées en décimaux. Associer chaque image à un tableau :



$$\begin{matrix}
 \mathbf{1} & \begin{pmatrix} 255 & 255 & 255 & 255 \\ 255 & 127 & 127 & 255 \\ 255 & 127 & 127 & 255 \\ 255 & 255 & 255 & 255 \end{pmatrix} & \mathbf{2} & \begin{pmatrix} 0 & 0 & 0 & 0 \\ 255 & 255 & 63 & 63 \\ 0 & 0 & 255 & 255 \\ 255 & 255 & 63 & 63 \end{pmatrix} & \mathbf{3} & \begin{pmatrix} 127 & 127 & 127 & 127 \\ 127 & 255 & 255 & 127 \\ 127 & 255 & 255 & 127 \\ 127 & 127 & 127 & 127 \end{pmatrix} & \mathbf{4} & \begin{pmatrix} 0 & 0 & 0 & 0 \\ 255 & 255 & 127 & 127 \\ 63 & 63 & 255 & 255 \\ 255 & 255 & 127 & 127 \end{pmatrix}
 \end{matrix}$$

- 3) Écrire un tableau de nombres correspondant à un assombrissement de l'image C.

Dans GIMP, produire une nouvelle image en 3x3 et en mode couleur RVB. Zoomer à 4000 %. La remplir de bleu. Exporter le fichier (« Exporter sous... ») dans votre dossier personnel au format « PPM » (en ASCII) sous le nom « bleu.ppm » (modifier l'extension en ppm à la main puis sélectionner ASCII). Ouvrir notepad++, charger le fichier « bleu.ppm » et vérifier que les pixels ont bien été codés en bleu (après l'entête).

- 4) Modifier le fichier dans le bloc note de manière à dessiner le drapeau français. « Enregistrer sous » avec le nom drapeau.ppm. Ouvrir le fichier « drapeau » dans Gimp pour vérifier (zoom à 4000 %).

Le CHR70M est un capteur qui se trouve dans certains appareils photo. Il totalise 70 millions de pixels sur une surface de 24 x 36 mm. Ses huit canaux de sortie cadencés à 30 MHz lui permettent de délivrer 3 images de 10000 x 7096 pixels par seconde.

Un pixel est constitué par quatre photosites : R, V, V, B, cela car l'œil humain est plus sensible au vert. Les photodétecteurs utilisés comme photosites ne sont pas sensibles aux couleurs : on place devant chacun un filtre (R, V ou B) pour traduire le signal électrique obtenu en information sur la couleur qui arrive en un point. Cette grille de filtres s'appelle la matrice de Bayer.

- 5) Calculer la taille d'un pixel, puis d'un photosite.
- 6) Calculer le débit binaire total en sortie du capteur en supposant un codage en 16 millions de couleurs.
- 7) Quelle est l'utilité de la matrice de Bayer ?
- 8) Comment la synthèse additive intervient-elle lors de la formation d'une image sur un écran ?
- 9) Est-ce que cette synthèse est aussi utilisée pour l'enregistrement d'une image ?