

# Codage des images

## 1 – IMAGES MATRICIELLE OU BITMAP

L'image est divisée en (**m colonnes** × **n lignes**) cellules appelées **pixel**. On associe alors une couleur à chaque pixel selon divers encodages.

La **définition** d'une image numérique correspond au **nombre de points (pixels) qui la composent**. La **résolution** d'une image est définie par un **nombre de pixels unité de longueur** de l'image numérique affichée ou numérisée c'est-à-dire la densité de pixel de l'image. La résolution s'exprime en **ppp (pixels par pouce)** ou en **dpi** en anglais (**dots per inch**). L'ensemble des deux définit la taille de l'image.

$$\text{Largeur} = \frac{\text{m colonnes}}{\text{Résolution}}$$

$$\text{hauteur} = \frac{\text{n lignes}}{\text{Résolution}}$$

### Exemple

L'image suivante possède une définition de **1900×1174** et une résolution de **72 ppp**. La taille de l'image est :

$$\text{Largeur} = \frac{1900}{72} = 26,4 \text{ pouces} = 67 \text{ cm}$$

$$\text{Hauteur} = \frac{1174}{72} = 16,3 \text{ pouces} = 41,4 \text{ cm}$$

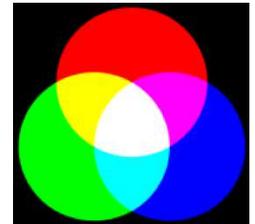


1. **Déterminer** le nombre de pixel d'une image **800×400**. **Déterminer** la largeur et la hauteur de cette image sachant qu'elle présente une résolution de **72 ppp**.
2. **Calculer** la résolution d'une image bitmap carrée de côté **10 cm** et de définition **800 × 800**.

## 2 – CODAGE DES COULEURS

Il existe plusieurs modes de codage informatique des couleurs, le plus utilisé pour le maniement des images est l'espace **colorimétrique Rouge, Vert, Bleu (RVB ou RVG : Red Green Blue) par synthèse additive**.

Une image **RVB** est composée de la somme des trois rayonnements lumineux **Rouge, Vert, Bleu** dont les **faisceaux sont superposés**. A l'**intensité maximale** ils produisent une **lumière blanche**. La gamme des couleurs reproductibles par ce mode, quoique conditionnée par la qualité du matériel employé, est très étendue, et reproduit bien les couleurs saturées. En contrepartie, elle convient mal à la restitution des nuances délicates des lumières intenses et des tons pastels.



Le codage de la couleur est réalisé sur **3 octets** dont les **valeurs codent la couleur** dans l'espace **RVB**. Chaque octet représente la valeur d'une composante couleur par un entier de **0 à 255**. Le nombre de couleurs différentes est de **256x256x256 = 16,8 Millions**. Une image numérique **RVB** est représentée par **3 tableaux à 2 dimensions** dont la taille dépend du nombre de pixels contenus dans l'image.

R	V	B	Couleur
0	0	0	Noir
0	0	1	Nuance de Noir
0	0	255	Bleu
0	255	0	Vert
255	0	0	Rouge
128	128	128	Gris
255	128	0	Orange
128	0	128	Violet
255	255	255	Blanc

Dans ce type d'image seul le **niveau de l'intensité est codé sur un octet (256 valeurs)**. Par convention, la **valeur 0 représente le noir** (intensité lumineuse nulle) et la **valeur 255 le blanc** (intensité lumineuse maximale) :

0	8	16	32	56	72	90	104	112	128
136	144	160	176	192	208	224	244	248	255

Pour passer d'une image couleur à une image en niveau de gris, on utilise la formule :

$$G = 0,299 \times R + 0,587 \times G + 0,114 \times B$$

Une image numérique en niveau de gris est représentée par **1 tableau à 2 dimensions** dont la taille dépend du nombre de pixels contenus dans l'image.

Les différents systèmes de colorimétrie :

Mode	Nb de bits par pixels	Nombre de couleurs	Remarques
<b>Monochrome ou Noir et Blanc</b>	<b>1</b>	<b>2</b>	Système utilisé pour <b>scanner les textes</b> pour faire de la reconnaissance de texte ( <b>OCR</b> )
<b>Niveaux de gris</b>	<b>8</b>	<b>256</b>	<b>Nuance de gris</b>
<b>Mode 4 bits ou 16 couleurs</b>	<b>4</b>	<b>16</b>	Palette de couleurs peu étendue réservée aux <b>dessins simples</b> sans couleurs nuancée
<b>Mode 8 bits ou 256 couleurs</b>	<b>8</b>	<b>256</b>	Palette de 256 couleurs qui permet de conserver une <b>taille raisonnable</b>
<b>Mode 16 bits</b>	<b>16</b>	<b>65536</b>	Palette de 64536 couleurs qui convient pour la <b>plupart des usages</b>
<b>Mode 24 bits ou Couleurs RVB</b>	<b>24</b>	<b>16,7 millions</b>	Mode <b>utilisé par défaut</b> par de nombreux logiciels
<b>Couleurs CMJN</b>	<b>32</b>	<b>4,3 Milliards</b>	<b>4 couleurs primaires</b> : Cyan, Magenta, Jaune et Noir (256 teintes). Utilisé par les <b>imprimantes</b>

1. **Indiquer** par combien de bits est codée chacune des 3 couleurs en mode couleurs 24 bits (ou couleurs vraies). **Donner** la valeur minimale et maximale de chacune des 3 composantes. **Déterminer** le nombre de nuances de couleurs obtenues avec ce type de codage couleur.
2. **Indiquer** quelle couleur est obtenue pour une intensité maximale des 3 couleurs RVB.
3. **Indiquer** quelle couleur est obtenue pour une intensité minimale des 3 couleurs RVB.
4. **Lancer** le logiciel **PAINT**, puis **ouvrir** l'image **Port.bmp**.
5. A l'aide du menu **Image/Attributs**, **compléter** la première ligne du tableau n°1 sur le Document Réponse.
6. **Enregistrer** l'image sous le nom **Port256.bmp** avec le type **Bitmap 256 couleurs**.
7. **Compléter** la deuxième ligne du tableau n°1 sur le Document Réponse.
8. **Ouvrir** à nouveau l'image **Port.bmp**. **Enregistrer-la** sous le nom **Port16.bmp** avec le type **Bitmap 16 couleurs**.
9. **Compléter** la troisième ligne du tableau n°1 sur le Document Réponse.
10. **Ouvrir** à nouveau l'image **Port.bmp**. **Enregistrez-la** sous le nom **Port2.bmp** avec le type **Bitmap Monochrome**.
11. **Complétez** la dernière ligne tableau n°1 sur le Document Réponse.
12. **Justifier** le lien entre les valeurs obtenues pour la dernière colonne et le nombre de bits par pixels.

### 3 – LES DIFFERENTS FORMATS D'IMAGES

Extension	Nombre de couleurs	Présentation
BMP	16,7 millions	<b>Standard Windows.</b> Format <b>très répandu</b> mais entraînant des fichiers <b>très volumineux</b> (très faible compression)
GIF	256	<b>Standard Internet</b> (excellente compression). Possibilité d' <b>affichage progressif, animation</b> ou de <b>zone réactive</b>
PNG	256 à 16,7 millions	Même type que le <b>format GIF</b> mais avec <b>des capacités de couleurs supérieures</b> . Format ouvert. Transparence possible.
PCX	16,7 millions	<b>Format standard</b> (Paintbrush) mais entraînant des fichiers <b>très volumineux</b>
TIF	16,7 millions	<b>Format reconnu</b> par l'ensemble des machines, très <b>utilisé en PAO</b> . Image de <b>qualité</b> mais fichiers <b>très volumineux</b>
JPEG (JPG)	16,7 millions	<b>Standard internet et photo.</b> Excellente compression avec plus ou moins de perte. Possibilité de zone réactive. <b>Choix du taux de compression</b> (donc de la <b>qualité de l'image</b> )

1. **Ouvrir** l'image **Paysage.bmp**.
2. A l'aide du menu **Image/Attributs**, **compléter** la première ligne du tableau n°2 sur le Document Réponse.
3. **Enregistrer** l'image sous le nom **PaysageGIF.gif** avec le type GIF.
4. **Compléter** la deuxième ligne du tableau n°2 sur le Document Réponse.
5. **Ouvrir** à nouveau l'image **Paysage.bmp**. **Enregistrer-la** sous **PaysagePG.jpg** avec le type JPEG.
6. **Complétez** la troisième ligne du tableau n°2 sur le Document Réponse.
7. **Justifier** pourquoi la qualité de l'image au format GIF est-elle altérée.
8. **Justifier** s'il y a une différence de qualité entre l'image JPEG et l'original au format BMP (**utiliser** éventuellement la fonction ZOOM).
9. **Déterminer** le temps nécessaire pour la transmission de chacune des images précédentes sur une ligne ADSL ayant un débit idéal de **512 ko/s**.
10. **Justifier** pourquoi le format JPG est un des formats les plus utilisés.

## 4 – CODAGE BINAIRE D'UNE IMAGE BMP

Le format **BMP** est un des formats d'image les plus simples. Il a été développé par Microsoft et IBM, ce qui explique qu'il soit particulièrement répandu sur les plateformes Windows. Un fichier **BITMAP** ou **BMP** est un fichier d'image graphique **stockant les pixels sous forme de tableau de points**. Le codage de l'image se fait en écrivant successivement les bits correspondant à chaque pixel, ligne par ligne.

La structure d'un fichier bitmap est la suivante :

- ❑ **en-tête du fichier** (en anglais file header) ;
- ❑ **en-tête du bitmap** (en anglais bitmap information header, appelé aussi information Header) ;
- ❑ **palette** (optionnelle) ;
- ❑ **corps de l'image**.

L'entête du fichier fournit des informations sur le type de fichier **BITMAP**, sa taille et indique où commencent les informations concernant l'image à proprement parler. L'entête est composé de **4 champs** :

- ❑ la **signature** (sur **2 octets**), indiquant le type de fichier **BMP** à l'aide des deux caractères. Pour une image **BMP Windows** : **BM (0x42 et 0x4D)** ;
- ❑ la **taille totale du fichier** en octets (codée sur **4 octets**). Les octets sont lus de droite à gauche ;
- ❑ un **champ réservé** (sur **4 octets**)
- ❑ l'**offset** ou **décalage de l'image** (sur **4 octets**) : **adresse relative** du début des informations concernant l'image par rapport au début du fichier.

L'entête de l'image fournit des **informations sur l'image**, notamment **ses dimensions et ses couleurs**.

L'entête de l'image est composé de **11 champs** :

- ❑ **taille de l'entête** de l'image en octets (codée sur **4 octets**). Pour une image **BMP Windows** : **0x28** ;
- ❑ **largeur de l'image** (sur **4 octets**) : **nombre de pixels horizontalement** (en anglais **width**) ;
- ❑ **hauteur de l'image** (sur **4 octets**) : **nombre de pixels verticalement** (en anglais **height**) ;
- ❑ **nombre de plans** (sur **2 octets**). Cette valeur a toujours pour valeur **1** ;
- ❑ **codage de la couleur** (sur **2 octets**) : **nombre de bits utilisés pour coder la couleur**. Cette valeur peut-être égale à **1, 4, 8, 16, 24** ou **32**.
- ❑ **méthode de compression** (sur **4 octets**) : **0** lorsque l'image **n'est pas compressée** ; **1, 2** ou **3** suivant le type de compression utilisé ;
- ❑ **taille totale de l'image** en octets (sur **4 octets**).
- ❑ **résolution horizontale** (sur **4 octets**) : nombre de pixels horizontaux par unité de longueur ;
- ❑ **résolution verticale** (sur **4 octets**) : nombre de pixels verticaux par unité de longueur ;
- ❑ **nombre de couleurs de la palette** (sur **4 octets**)
- ❑ **nombre de couleurs importantes** de la palette (sur **4 octets**). Valeur à **0** lorsque chaque couleur a son importance.

1. **Ouvrir** l'image **aeroport.bmp**. **Déterminer** à partir de ses propriétés la taille en octets du fichier ainsi que la définition de cette image (nombre de pixels verticaux et nombre de pixels horizontaux).
2. **Lancer** le logiciel **Free hex Editor**. **Ouvrir** le fichier **aeroport.bmp**.
3. **Justifier** la valeur des octets numéros **0x0000** et **0x0001**.
4. **Déterminer** la taille en octet du fichier **aeroport.bmp** à partir de l'entête du fichier. **Vérifier** que cette taille correspond à celle trouvée à la question 1.
5. **Déterminer** le numéro de l'octet à partir commence le codage de l'image.
6. **Donner** le nombre de pixels horizontaux et le nombre de pixels verticaux. **Vérifier** que ces valeurs correspondent aux valeurs trouvées à la question 1.
7. **Donner** le numéro de l'octet et la valeur permettant de justifier que le codage des couleurs utilisé est le codage sur **24 bits**.
8. **Donner** le numéro de l'octet et la valeur permettant de justifier que l'image n'est pas compressée.
  
9. **Ouvrir** l'image **carre.bmp**. **Déterminer** à partir de ses propriétés la définition de l'image (nombre de pixels verticaux et nombre de pixels horizontaux).
10. **Calculer** la taille de l'image sachant que le codage est le codage **RVB**.
11. **Lancer** le logiciel **Free hex Editor**. **Ouvrir** le fichier **carre.bmp**.
12. **Déterminer** la taille de l'image **carre.bmp** à partir de l'entête de l'image. **Vérifier** que cette taille correspond à celle trouvée à la question 10.
13. **Donner** la valeur des **3 octets** codant le **1<sup>er</sup> pixel**.
14. **Remplacer** ces **3 octets** par les valeurs suivantes : **0xFF, 0x00, 0x00**. **Enregistrer** le fichier.
15. **Ouvrir** le fichier **carre.bmp** avec le logiciel **Paint**. **Préciser** quel pixel a été modifié. **Donner** la nouvelle couleur du pixel.
16. **Ouvrir** à nouveau le fichier **carre.bmp** avec le logiciel **EditHexa**
17. **Remplacer** les **3 octets** codant le **2<sup>nd</sup> pixel** par les valeurs suivantes : **0x00, 0xFF, 0x00**. **Enregistrer** le fichier.
18. **Ouvrir** le fichier **carre.bmp** avec le logiciel **Paint**. **Donner** la nouvelle couleur du pixel.
19. **Ouvrir** à nouveau le fichier **carre.bmp** avec le logiciel **Free hex Editor**.
20. **Remplacer** les **3 octets** codant le **3<sup>ième</sup> pixel** par les valeurs suivantes : **0x00, 0x00, 0xFF**. **Enregistrer** le fichier.
21. **Ouvrir** le fichier **carre.bmp** avec le logiciel **Paint**. **Donner** la nouvelle couleur du pixel.
22. **Expliquer** comment est réalisé le codage de l'image en expliquant dans quel ordre sont codés les pixels et les couleurs.

# DOCUMENT REPONSE

Image	Nombre de pixel	Nb max de couleurs	Taille fichier en ko	Nb de bits par pixel	Qualité perçue de l'image	Rapport taille fichier / taille minimum*
<i>Port.bmp</i>						
<i>Port256.bmp</i>						
<i>Port16.bmp</i>						
<i>Port2.bmp</i>						

\* ne remplissez cette colonne qu'à la fin, après avoir repéré le plus petit des fichiers, et arrondissez le résultat.

Image	Nombre de pixel	Taille fichier en ko	Qualité perçue de l'image
<i>Paysage.bmp</i>			
<i>Paysage GIF.gif</i>			
<i>PaysageJPG.jpg</i>			