

DIC	OTSCIS	MSOST	IP
Séance 2 Seq16	Comment un message informatique est-il codé et envoyé ?		Technologie Cycle 4 - 4ème
IP 1.2 IP 4.2 IP 5.5	Organiser, structurer et stocker des ressources numériques.		

Je viens de répondre ce message à un ami :

Nous allons tenter de savoir comment ce message est codé et envoyé par le téléphone (→ Ce SMS est composé d'un texte, d'un Smiley et il faut l'envoyer à un destinataire précis !)



II) Comment une image est-elle codée par un ordinateur ?

Nous savons qu'avec **un octet** (ensemble de 8 bits), on pouvait coder **256 possibilités**. Pour du texte, c'est suffisant, mais qu'en est-il pour une image ?

Si vous n'avez pas photofiltre sur ton ordinateur, il est téléchargeable ici :

https://www.01net.com/telecharger/windows/Multimedia/photo_numerique/fiches/telecharger-21756.html

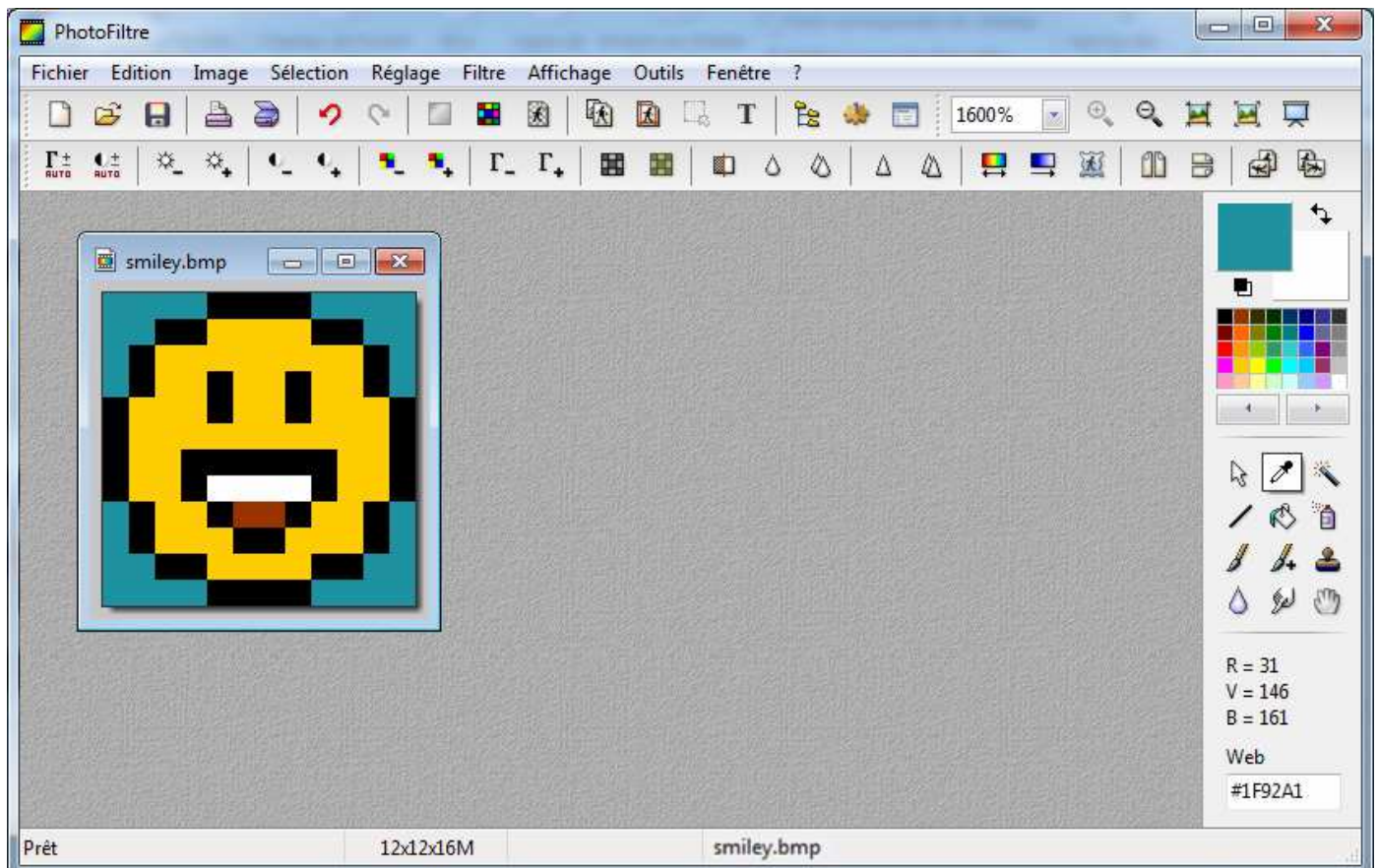
Sur ordinateur, ouvrez avec Photofiltre le fichier « smiley.bmp » disponible en fichier joint.

- L'image est très petite, zoomez au maximum.

① Sur cette image : (pas besoin de compter, c'est écrit quelque part dans la fenêtre !)

- Combien y a-t-il de pixels en largeur ? 12
- Combien y a-t-il de pixels en hauteur ? 12
- Combien y a-t-il de pixels en tout ? $12 \times 12 = 144$
- Où trouve-t-on ces informations ? En bas de la fenêtre (12x12x16M)

- Cliquez sur l'outil « pipette ». Cliquez sur le premier Pixel en haut à gauche de l'image.



② D'après-vous, que signifient les 3 valeurs 31, 146 et 161 de R, V et B en dessous des icônes d'outils ?

niveau de Rouge 31, niveau de Vert 146, niveau de Bleu, 161

③ À l'aide des documents des séances précédentes, donnez le codage en binaire de la couleur qui entoure le

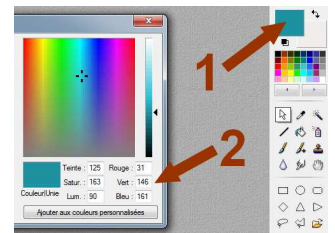
smiley : 1F92A1 : 0001 1111 1001 1011 0001

④ Donnez les valeurs en hexadécimal des quatre autres couleurs qui composent le smiley :

Noir : # 000 000 Blanc : # FFFF FFFF Jaune : # FFCC00 Marron : # 993300

⑤ Remplacez la couleur verte du fond par la couleur suivante :

R = 158 V = 158 B = 158



⑥ Quelle est cette couleur ? Pourquoi ? Gris, mélange de rouge, vert et bleu aux mêmes proportions

⑦ Mettez le vert à 0, quelles couleur obtient-on et pourquoi ?

Violet

⑧ Combien faut-il d'octet pour coder la couleur d'un pixel ? 3 octets

⑨ A l'aide de la réponse à la question ①, indiquez combien faut-il d'octets au minimum pour coder l'ensemble des

pixels de l'image (précisez le calcul) : $144 \times 8 = 432$ octets

Voici à quoi ressemble le SMS envoyé à notre ami :

M d r !

```

01001101 01100100 01110010 00100001
10100001 10010010 00100001 10100001 10010010 00100001 10100001 10010010 00100001 10100001 10010010 00100001
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
10100001 10010010 00100001 10100001 10010010 00100001 10100001 10010010 00100001 10100001 10010010 00100001
10100001 10010010 00100001 10100001 10010010 00100001 00000000 00000000 00000000 00000000 00000000 00000000
0000 1 00000000 11001100 11111111 00000000 11001100 11111111 00000000 11001100 11111111
0000 =161 =146 =31 0 00000000 00000000 00000000 10100001 10010010 00100001 10100001 10010010 00100001
1010 1 00000000 00000000 00000000 00000000 00000000 11001100 11111111 00000000 11001100 11111111
00000000 11001100 11111111 00000000 00000000 00000000 00000000 00000000 00000000 00000000 11001100 11111111
00000000 11001100 11111111 00000000 11001100 11111111 00000000 00000000 00000000 10100001 10010010 00100001
10100001 10010010 00100001 00000000 00000000 00000000 00000000 11001100 11111111 00000000 11001100 11111111
00000000 00000000 00000000 00000000 00110011 10011001 00000000 00110011 10011001 00000000 00000000 00000000
00000000 11001100 11111111 00000000 11001100 11111111 00000000 00000000 00000000 10100001 10010010 00100001
00000000 00000000 00000000 00000000 11001100 11111111 00000000 11001100 11111111 00000000 00000000 00000000
11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
00000000 00000000 00000000 00000000 1001100 11111111 00000000 11001100 11111111 00000000 00000000 00000000
00000000 00000000 00000000 00000000 11001100 11111111 00000000 11001100 11111111 00000000 00000000 00000000
00000000 00000000 00000000 00000000 11001100 11111111 00000000 11001100 11111111 00000000 11001100 11111111
00000000 00000000 00000000 00000000 11001100 11111111 00000000 11001100 11111111 00000000 11001100 11111111
00000000 11001100 11111111 00000000 11001100 11111111 00000000 11001100 11111111 00000000 00000000 00000000
10100001 10010010 00100001 00000000 00000000 00000000 00000000 11001100 11111111 00000000 11001100 11111111
00000000 00000000 00000000 00000000 11001100 11111111 00000000 11001100 11111111 00000000 00000000 00000000
00000000 11001100 11111111 00000000 11001100 11111111 00000000 00000000 00000000 10100001 10010010 00100001
10100001 10010010 00100001 00000000 00000000 00000000 00000000 11001100 11111111 00000000 11001100 11111111
00000000 11001100 11111111 00000000 11001100 11111111 00000000 11001100 11111111 00000000 11001100 11111111
00000000 11001100 11111111 00000000 11001100 11111111 00000000 00000000 10100001 10010010 00100001
10100001 10010010 00100001 10100001 10010010 00100001 00000000 00000000 00000000 00000000 00000000 00000000
00000000 11001100 11111111 00000000 11001100 11111111 00000000 11001100 11111111 00000000 11001100 11111111
00000000 00000000 00000000 00000000 00000000 00000000 10100001 10010010 00100001 10100001 10010010 00100001
10100001 10010010 00100001 10100001 10010010 00100001 10100001 10010010 00100001 10100001 10010010 00100001
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
10100001 10010010 00100001 10100001 10010010 00100001 10100001 10010010 00100001 10100001 10010010 00100001

```

On remarque que le texte « Mdr! » n'est codé que sur les 4 premiers octets alors que l'image est codé sur 432 octets ! Les images prennent donc beaucoup plus de place qu'un texte.
 Dans la réalité, avant le codage de pixels, on a 54 octets supplémentaires pour les informations sur l'image (type, taille, etc...). L'image fait donc 432 + 54 = 486 octets

smiley.bmp
 Type d'élément : Fichier BMP
 Dimensions : 12 x 12
 Taille : 486 octet(s)

- 1) 12 pixels en largeur. 12 pixels en hauteur. 12x12=144 pixels en tout. Ces informations sont en bas de la fenêtre (12x12x16M)
- 2) niveau de Rouge 31, niveau de Vert 146, niveau de Bleu, 161
- 3) 1F92A1 : 0001 1111 1001 1011 0001
- 4) Noir : # 000 000 Blanc : # FFFF FFFF Jaune : # FFCC00 Marron : # 993300
- 6) Gris, mélange de rouge, vert et bleu aux mêmes proportions
- 7) Violet
- 8) 3 octets
- 9) 144 x 8 = 432 octets