



Comment coder l'information ?

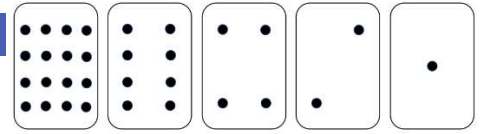
Nicolas TOURREAU @IANum_Techno – Cité Scolaire de Lannemezan

Nom Prénom :

Classe :

Coder des chiffres

À partir des cartes distribuées, répondre aux questions :



Remarques à propos du nombre de points qui figure sur les cartes : _____

Combien de points devrait avoir la prochaine carte à gauche ? _____ et la suivante... ? _____

Utiliser ces cartes pour représenter des nombres. Pour cela retourner certaines et additionner les points qui restent visibles. Afficher 6 (cartes 4 points et 2 points), puis 15 (cartes 8, 4, 2 et 1 points), puis 21 (16, 4 et 1).

Lorsqu'une carte d'un nombre binaire n'est pas visible, elle est représentée par un 0. Lorsqu'elle est visible, elle est représentée par un 1. C'est le système de **numération binaire**, contrairement à l'écriture des nombres du **système décimal** qui utilise les chiffres de 0 à 9.

Trouver comment obtenir 3, 12, 19. Existe-t-il plusieurs moyens d'obtenir un nombre ? _____

Quel est le plus grand nombre que l'on peut obtenir ? _____ Quel est le plus petit ? _____

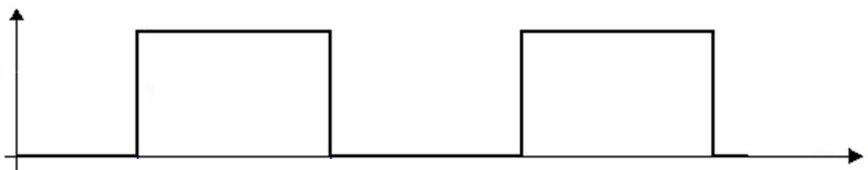
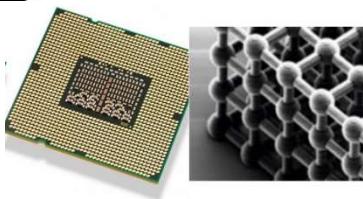
Y a-t-il un nombre compris entre le plus grand et le plus petit que l'on ne puisse pas obtenir ? _____

Inversement, trouver combien fait 10111 ? _____ 11101 ? _____

Le code binaire



À partir de la vidéo jusqu'à la 5' min expliquer d'où vient le codage binaire :



Explications : _____

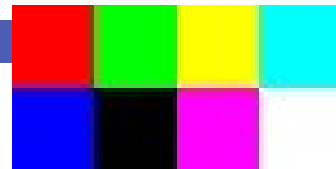
S				S				N				T			

Coder une image



Ouvrir le fichiers 8px.bmp à partir de l'application en ligne [HexEd.it](https://hexed.it)

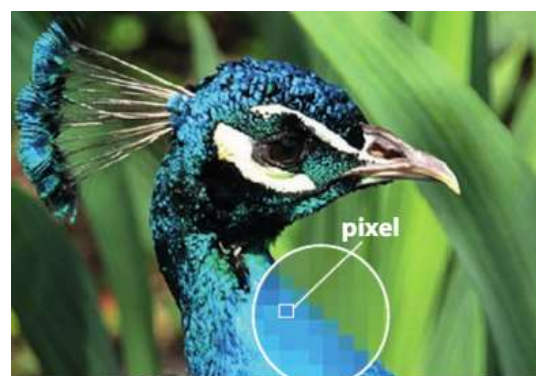
Noter les résultats obtenus :



 En-tête Valeur constante : _____ octets	Taille (octets)		Code (Hexadécimal)			Signification	
	42		4D			Caractères B(42) et M(4D) indiquant le _____ fichier. Ici _____	
			00		00		_____ du fichier 0000004E = _____ octets
	00		00		00		Réservé (toujours à 0)
			00		00		Offset ou Décalage de l'image = _____ octets (début des informations concernant l'image par rapport au début du fichier)
			00		00		Taille de l'entête = _____ octets
			00		00		_____ de l'image = _____ pixels
			00		00		_____ de l'image = _____ pixels
			00				Nombre de plans utilisés = ____ (Cette valeur vaut toujours 1)
			00				Nombre de ____ par pixel = ____ soit ____ octets (1, 4, 8, 16, 24 ou 32)
	00		00		00		Méthode de compression : 0 pas de compression
	00		00		00		Taille de l'image 00000018 = 24 octets = 8 (pixels) x 3 (octets par pixel)
					00		Résolution horizontale = _____ pixels par mètre
					00		Résolution verticale = _____ pixels par mètre
	00		00		00		Couleurs utilisées : 0 palette entière
	00		00		00		Nombre de couleurs important (Ce champ peut être égal à 0 lorsque chaque couleur a son importance)
	Code Image Largeur (px) X Hauteur (px) : _____ pixels Soit _____ octets	FF		00		00	
00		00		00			
FF		00		FF			
FF		FF		FF			
00		00		FF			
00		FF		00			
00		FF		FF			
FF		FF		00			
Total : _____ octets							

Coder une couleur

D'après les résultats précédents et l'image ci-dessous, comment semble être codée une image ?



Comment sont codées chacune des couleurs pour chaque pixel ?

Sur quel principe scientifique se base ce codage ?

Avec un codage sur 24 bits ou 3 octets combien de nuances de couleurs sont possibles ?

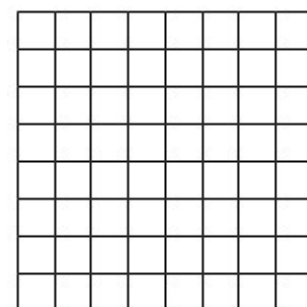
Dans le code d'une image, dans quel sens se fait la lecture de l'image ?

Exercice d'application : On donne ci-dessous le code du fichier correspondant à une image de 8 x 8 pixels au format bmp.

Quelle est la taille du fichier ? _____ Combien d'octets sont utilisés pour coder les pixels ? _____

Colorier les pixels pour reconstituer l'image :

Information sur le fichier		-Sans nom-	Sans titre.bmp	smiley.bmp
Nom du fichier	smiley.bmp	00000000	42 4D F6 00 00 00 00 00	00 00 36 00 00 00 28 00
Taille du fichier	246 octets	00000010	00 00 08 00 00 00 08 00	00 00 01 00 18 00 00 00
Inspecteur de données (Petit-boutiste)		00000020	00 00 00 00 00 00 20 1C	00 00 20 1C 00 00 00 00
Type	Non signé (+) Signé (±)	00000030	00 00 00 00 00 00 FF FF	FF FF FF FF 00 00 00 00
Type de caractères	Données invalides	00000040	00 00 00 00 00 00 00 00	FF FF FF FF FF FF FF FF
UTF-8		00000050	FF 00 00 00 00 FF FF 00	FF FF 00 FF FF 00 FF FF
Binaire	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	00000060	00 00 00 FF FF FF 00 00	00 00 FF FF 00 FF FF 00
Inspecteur de données (Gros-boutiste)		00000070	00 00 00 00 00 00 FF FF	00 FF FF 00 00 00 00 00
		00000080	00 00 FF FF 00 FF FF 00	FF FF 00 FF FF 00 FF FF
		00000090	00 FF FF 00 00 00 00 00	00 00 FF FF 00 00 00 00
		000000A0	FF FF 00 FF FF 00 00 00	00 FF FF 00 00 00 00 00
		000000B0	00 00 FF FF 00 FF FF 00	FF FF 00 FF FF 00 FF FF
		000000C0	00 FF FF 00 00 00 FF FF	FF 00 00 00 00 FF FF 00
		000000D0	FF FF 00 FF FF 00 FF FF	00 00 00 FF FF FF FF FF
		000000E0	FF FF FF FF 00 00 00 00	00 00 00 00 00 00 00 00
		000000F0	FF FF FF FF FF FF +	



Un autre code pour créer des images...

Ouvrir les fichiers *Plan-Lyon-Metro-Tramway.bmp* et *Plan-Lyon-Metro-Tramway.svg* avec le navigateur Firefox.
Zoomer à 300% (CTRL + Molette souris).

Remarques : _____

Dans Firefox, faire un clic droit sur chaque image. Les options disponibles sont-elles les mêmes ? Cliquer sur afficher le code source pour l'image .svg. Comment a été créé l'image en svg ?

Avantage du format svg ? _____

Copier le code dans le BlocNote et enregistrer sous le nom *cercle.svg*. Ouvrir le fichier avec le navigateur Firefox :

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<svg width="800px" height="800px" xmlns="http://www.w3.org/2000/svg">
<circle cx="400px" cy="400px" r="100px" fill="black" />
<title> Disque noir en SVG </title>
<desc> <Creator>Prenom NOM</Creator> </desc></svg >
```

