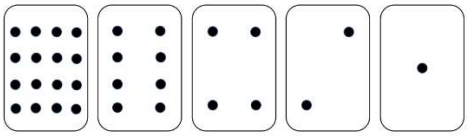


# Comment coder l'information ?

*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 ? _____

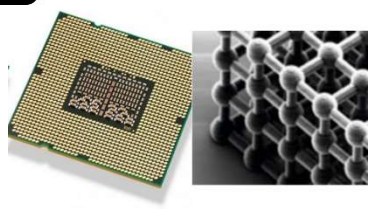
Quel est le plus grand nombre que l'on peut obtenir ? _____ Quel est le plus petit ? _____

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 : _____

_____

_____

_____

_____

## Coder des caractères

Ouvrir les fichiers snt.txt et snt.doc dans un éditeur de texte.

Remarques : _____

_____

_____

Ouvrir les fichiers snt.txt et snt.doc à partir de l'application en ligne [hexed.it](hexed.it)

Remarques : _____

_____

_____

Quelle est la taille de chaque fichier ?







Quelle est l'unité ? _____

D'où vient ce mot ? _____

Comment expliquer la différence de taille ?

Explications : _____

_____

_____

_____

Qu'est-ce que le code ASCII ? _____

_____

_____

_____

_____

_____

_____

## ASCII TABLE

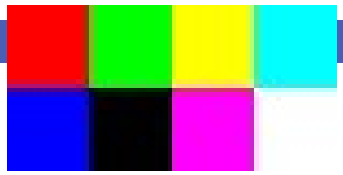
| Decimal | Hex | Char | Decimal | Hex | Char | Decimal | Hex | Char | Decimal | Hex | Char |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | [NULL] | 32 | 20 | [SPACE] | 64 | 40 | @ | 96 | 60 | ` |
| 1 | 1 | [START OF HEADING] | 33 | 21 | ! | 65 | 41 | A | 97 | 61 | a |
| 2 | 2 | [START OF TEXT] | 34 | 22 | " | 66 | 42 | B | 98 | 62 | b |
| 3 | 3 | [END OF TEXT] | 35 | 23 | # | 67 | 43 | C | 99 | 63 | c |
| 4 | 4 | [END OF TRANSMISSION] | 36 | 24 | $ | 68 | 44 | D | 100 | 64 | d |
| 5 | 5 | [ENQUIRY] | 37 | 25 | % | 69 | 45 | E | 101 | 65 | e |
| 6 | 6 | [ACKNOWLEDGE] | 38 | 26 | & | 70 | 46 | F | 102 | 66 | f |
| 7 | 7 | [BELL] | 39 | 27 | ' | 71 | 47 | G | 103 | 67 | g |
| 8 | 8 | [BACKSPACE] | 40 | 28 | ( | 72 | 48 | H | 104 | 68 | h |
| 9 | 9 | [HORIZONTAL TAB] | 41 | 29 | ) | 73 | 49 | I | 105 | 69 | i |
| 10 | A | [LINE FEED] | 42 | 2A | * | 74 | 4A | J | 106 | 6A | j |
| 11 | B | [VERTICAL TAB] | 43 | 2B | + | 75 | 4B | K | 107 | 6B | k |
| 12 | C | [FORM FEED] | 44 | 2C | , | 76 | 4C | L | 108 | 6C | l |
| 13 | D | [CARRIAGE RETURN] | 45 | 2D | - | 77 | 4D | M | 109 | 6D | m |
| 14 | E | [SHIFT OUT] | 46 | 2E | . | 78 | 4E | N | 110 | 6E | n |
| 15 | F | [SHIFT IN] | 47 | 2F | / | 79 | 4F | O | 111 | 6F | o |
| 16 | 10 | [DATA LINK ESCAPE] | 48 | 30 | 0 | 80 | 50 | P | 112 | 70 | p |
| 17 | 11 | [DEVICE CONTROL 1] | 49 | 31 | 1 | 81 | 51 | Q | 113 | 71 | q |
| 18 | 12 | [DEVICE CONTROL 2] | 50 | 32 | 2 | 82 | 52 | R | 114 | 72 | r |
| 19 | 13 | [DEVICE CONTROL 3] | 51 | 33 | 3 | 83 | 53 | S | 115 | 73 | s |
| 20 | 14 | [DEVICE CONTROL 4] | 52 | 34 | 4 | 84 | 54 | T | 116 | 74 | t |
| 21 | 15 | [NEGATIVE ACKNOWLEDGE] | 53 | 35 | 5 | 85 | 55 | U | 117 | 75 | u |
| 22 | 16 | [SYNCHRONOUS IDLE] | 54 | 36 | 6 | 86 | 56 | V | 118 | 76 | v |
| 23 | 17 | [ENG OF TRANS. BLOCK] | 55 | 37 | 7 | 87 | 57 | W | 119 | 77 | w |
| 24 | 18 | [CANCEL] | 56 | 38 | 8 | 88 | 58 | X | 120 | 78 | x |
| 25 | 19 | [END OF MEDIUM] | 57 | 39 | 9 | 89 | 59 | Y | 121 | 79 | y |
| 26 | 1A | [SUBSTITUTE] | 58 | 3A | : | 90 | 5A | Z | 122 | 7A | z |
| 27 | 1B | [ESCAPE] | 59 | 3B | ; | 91 | 5B | [ | 123 | 7B | { |
| 28 | 1C | [FILE SEPARATOR] | 60 | 3C | < | 92 | 5C | \ | 124 | 7C | \| |
| 29 | 1D | [GROUP SEPARATOR] | 61 | 3D | = | 93 | 5D | ] | 125 | 7D | } |
| 30 | 1E | [RECORD SEPARATOR] | 62 | 3E | > | 94 | 5E | ^ | 126 | 7E | ~ |
| 31 | 1F | [UNIT SEPARATOR] | 63 | 3F | ? | 95 | 5F | _ | 127 | 7F | [DEL] |

| Decimal (Base 10) | Binary (Base 2) | Hexadecimal (Base 16) |
|---|---|---|
| 0 | 0000 | 0 |
| 1 | 0001 | 1 |
| 2 | 0010 | 2 |
| 3 | 0011 | 3 |
| 4 | 0100 | 4 |
| 5 | 0101 | 5 |
| 6 | 0110 | 6 |
| 7 | 0111 | 7 |
| 8 | 1000 | 8 |
| 9 | 1001 | 9 |
| 10 | 1010 | A |
| 11 | 1011 | B |
| 12 | 1100 | C |
| 13 | 1101 | D |
| 14 | 1110 | E |
| 15 | 1111 | F |

Quel est le code ASCII pour le « s » minuscule, le « S » « N » et « T » majuscules :

| s | S | N | T |
|---|---|---|---|
| | | | |
| | | | |

## Coder une image



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

Noter les résultats obtenus :

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

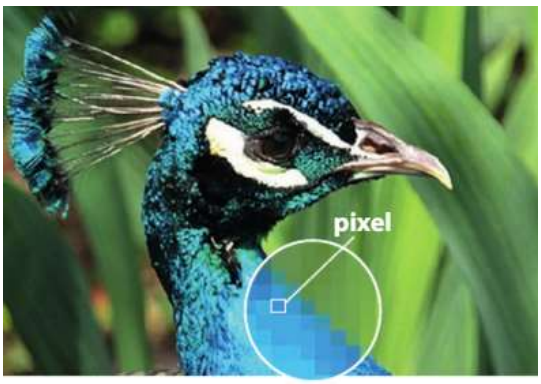
_____

_____

_____

_____

_____

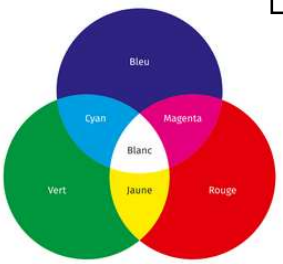


Comment sont codées chacune des couleurs pour chaque pixel ?

R ☐ V ☐ B ☐

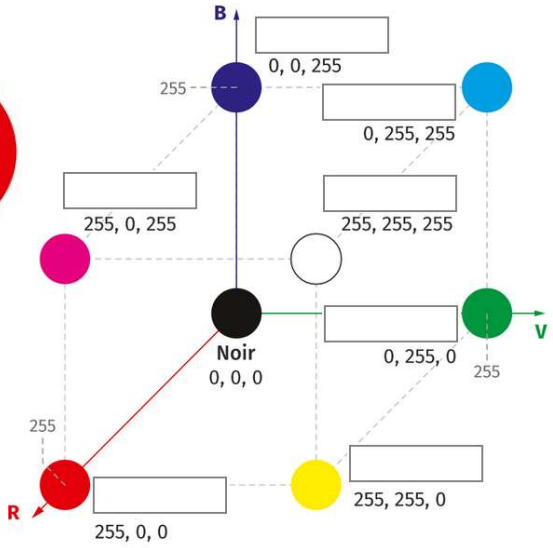
Sur quel principe scientifique se base ce codage ?

_____




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**

| | |
|---|---|
| Nom du fichier | smiley.bmp |
| Taille du fichier | 246 octets |

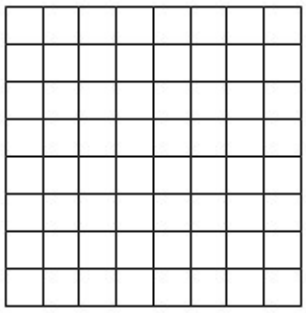
**Inspecteur de données (Petit-boutiste)**

| Type | Non signé (+) | Signé (±) |
|---|---|---|
| Type de caractères UTF-8 | Données invalides | |
| Binaire | | |

**Inspecteur de données (Gros-boutiste)**

-Sans nom- × Sans titre.bmp × smiley.bmp ×

```
00000000  42 4D F6 00 00 00 00 00 00 00 36 00 00 00 28 00
00000010  00 00 08 00 00 00 08 00 00 00 01 00 18 00 00 00
00000020  00 00 00 00 00 00 20 1C 00 00 20 1C 00 00 00 00
00000030  00 00 00 00 00 00 FF FF FF FF FF 00 00 00 00 00
00000040  00 00 00 00 00 00 00 00 FF FF FF FF FF FF FF FF
00000050  FF 00 00 00 00 FF FF 00 FF FF 00 FF FF 00 FF FF
00000060  00 00 00 FF FF FF 00 00 00 00 FF FF 00 FF FF 00
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 >
```