NSI 01

LES NOMBRES ENTIERS NON SIGNÉS

[Stéphane BEAUDET – Frédéric PEURIERE]

Savoir comment sont représentées les données dans un ordinateur.

Connaître la base 2 et savoir la manipuler : évaluer le nombre de bits nécessaires pour l'écriture d'un entier en base 2, la somme de deux entiers ou le produit de deux entiers.

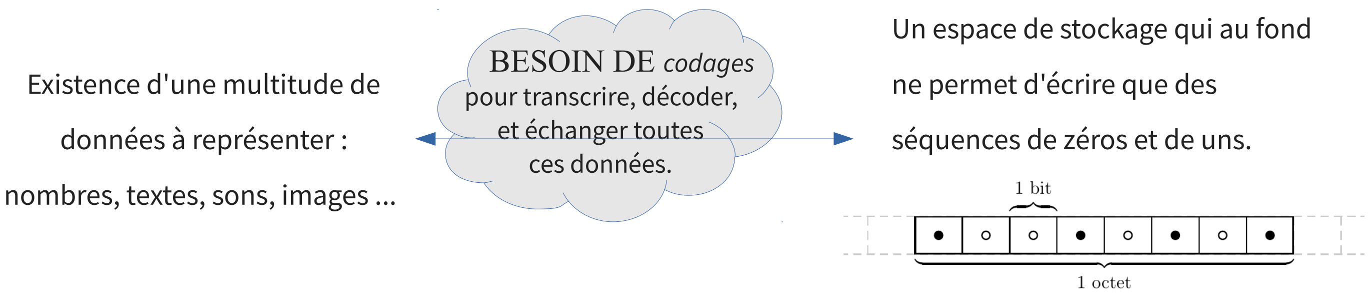
Connaître l'écriture d'un entier dans la base 2, 10 et 16 : savoir convertir une base dans une autre

→ les ordinateurs sont BINAIRES!

Lorsqu'un ordinateur traite du texte, du son, de l'image, de la vidéo, il traite en réalité exclusivement des **informations binaires.**

C'est une information qui ne peut avoir que deux états (chargé – non chargé, haut – bas, troué – non troué) : la mémoire vive (la « RAM ») est formée de millions de composants électroniques qui peuvent retenir ou relâcher une charge électrique : la coutume veut qu'on symbolise une information binaire sous la forme de 1 et de 0.

Ce n'est là qu'une **représentation**



1. Numériser des informations

ACTIVITÉ : Jouons au jeu du portrait.

Devinez le nom d’une personne en ne posant que des questions auxquelles on répond par oui ou par non dans la liste suivante

QUI ?	Genre ?	Look EMO?	Moins de 15 ans ?	Code
Pierre	garçon	Oh non ! 0	oui	‘001’
Nadia	fille	Good 1	oui	‘111’
David	garçon	Enorme 1	non	‘010’
Hamdi	garçon	Certes 1	oui	
Marie	fille	Surtout pas 0	non	
Adèle		Evidemment		‘110’
Ming Fu		Beurk		‘101’
Igor	garçon	Horreur	non	

Pour que les choix restent binaires, la réponse à propos du « Look EMO » ne peut être que oui ou non , sans introduire les nuances explicitées ici.

1°) Compléter le tableau ci dessus en faisant en sorte que deux personnes ne puissent avoir les mêmes attributs dans la liste proposée .
2°) Combien faut-il au minimum de questions oui/non pour deviner une personne de ce tableau? Donnez un contre exemple du fait que 2 bits suffisent (montrez donc que 2 bits ne suffisent pas....)
3°) Réciproquement : peut on toujours distinguer 8 éléments avec 3 bits d’information ? Expliquez. Donnez les raisons pour lesquelles c’est possible ici
4°) Si la réponse à propos du « Look EMO » avait été explicitée comme dans le tableau et non binaire, que se serait-il passé ?
5°) Pour deviner une personne parmi seize, combien faudrait-il de questions binaires au minimum? Quelles propriétés devraient avoir ces questions pour être en nombre minimum?

ALLER PLUS LOIN Refaire le jeu avec quatre personnes de la classe en leur trouvant des bits d’information pertinents.

2. Numériser des nombres

2.1. Coder des nombres en ….

2.1.1 « Rappel » : la numération de <u>position</u> en <u>base décimale</u>

Rappelez les 10 SYMBOLES de la base 10 :………..…………………………………
Avec ces derniers, on peut compter jusqu'à 9. Comment faire pour aller au-delà de 9 ?……………………
………………………………………………………………………………………………………
Ainsi en base 10 : $(135)_{10}=100+30+5=1\times100+3\times10+5\times1=1\times10^2+3\times10^1+5\times10^0$

(5 est le chiffre des……………… ; 3 est le chiffre des ………………. et 1 celui des ………………….)

Cette écriture est différente de $(351)_{10}=$...
bien que composée des mêmes chiffres

2.1.2 La base 2

Combien y a t il de symboles : … : … et...... (chiffres binaires ou *Binary Digits*, plus simplement ***bits***)
Avec ces derniers, on peut compter jusqu'à ….. Et si l'on veut aller au-delà, il faut changer de rang (base **BINAIRE**)
En travaillant comme en base 10, compléter : $(1011)_2=....\times2^3+0\times2^{...}+.....\times........+1\times2^0=(....)_{10}$
On reprend les même chiffres : est ce la même valeur que $(1110)_2$? Expliquez.......................................
…..

2.1.3 La base 16

Il faut …..... SYMBOLES ici 10 chiffres 0,1,2,3,4,5,6,7,8,9 (...il n’y a pas d’autres chiffres!) et 6 lettres : A,B,C,D,E,F (qui correspondent en décimal respectivement à 10,11,12,13,14,15)
Avec ces derniers, on peut compter jusqu'à …....... Et si l'on veut aller au-delà, il faut changer de rang (base HEXADECIMALE)
Exemple :
$(2A4D)_{16}=2\times16^3+10\times16^2+4\times16^1+13\times16^0=(10829)_{10}\neq(AD42)_{16}$ *encore la position...*

Il existe donc plusieurs bases pour écrire des nombres. Pour exprimer un nombre **n** dans une base **b**, il suffit de l’écrire comme ceci :
soit b un entier supérieur ou égal à 2, pour tout entier n de $\mathbb{N}$, il existe p dans $\mathbb{N}$ et $(a_0,a_{1,...},a_p)$ des entiers compris entre 0 et b-1 tels que

$$n = (a_0a_1...a_p)_b=a_p\times b^p+...+a_1\times b^1+...a_0\times b^0=\sum_{i=0}^{p}a_i\times b^i$$

<u>Définition</u> : a_p s’appelle le chiffre de POIDS FORT ; a_0 celui de POIDS FAIBLE (voir boutisme)

Il faut apprendre à passer d’une base à l’autre : <u>TD méthode conversions</u>

2.2 Encodage

Un ordinateur stocke donc en binaire des informations sous forme de paquets de 8 informations appelés **mots** ou ***octet , byte*** en anglais (les processeurs récents disposent de 32 – soit 4 octets- ou 64 bits pour stocker UN nombre…

Compléter le tableau

	Calcul des possibles	**Les entiers de...à…**
UN OCTET		
16 BITS		
32 BITS		
64 BITS		

Mais il faudra aussi stocker des (approximations de) nombres réels, des caractères alpha-numériques, des textes, du son ou des images : il faut donc inventer des ENCODAGES pour représenter ces informations

Différents types existent donc : **`char`**, **`int`**, **`unsigned int`**, **`long`** ...(plus tard...)

2.2.1 Entier positif en binaire sur n bits

On utilise l'écriture binaire vue en début de chapitre : si le nombre de bits est inférieur à n, on rajoute des 0 pour parvenir à n bits. S'il n'y a pas assez de bits disponibles, on dit que la capacité d'écriture est insuffisante (*« overflow »*)

Exemple :

Entier	Sur 4 bits	Sur 8 bits	Sur 16 bits	Sur 32 bits
135	…………….	10000111	…………………………	00000000000000000000000010000111

2.2.2 Somme et produits

Compléter la table d'addition binaire :

+	0	1
0		
1		10

Aussi pour sommer 2 nombres en binaire, il suffit d'additionner les 1 ou les 0 un à un, en démarrant par la droite :

$$(135)_{10}+(100)_{10}=(10000111)_2+(1100100)_2$$

donc

```
   10000111
 +  1100100
 = ..........................
```

et $(......................)_2=(235)_{10}$ C'est le bon résultat !!

Compléter la table de multiplication binaire :

+	0	1
0		
1		

Aussi pour multiplier 2 nombres en binaire, il suffit de procéder comme en base 10 et de décaler d'un rang quand on change le bit du multiplicateur :

$$(6)_{10}\times(3)_{10}=(110)_2\times(11)_2$$

donc

```
    00110
  x 00011
→   .....
  + 0110
  = ..............
```

et $(..............)_2=(18)_{10}$ Ok.