Complément : Binaire et hexadécimal

Réfléchissons un peu sur notre façon d'écrire les nombres !

Depuis le Moyen Âge, on écrit les nombres entiers naturels en *notation décimale à position (ou notation à position en base dix)*. Cela signifie que, pour écrire le nombre entier naturel *n*, on commence par imaginer *n* objets, que l'on groupe par paquets de dix, puis on groupe ces paquets de dix objets en paquets de dix paquets, etc. À la fin, il reste entre zéro et neuf objets isolés, entre zéro et neuf paquets isolés de dix objets, entre zéro et neuf paquets isolés de cent, etc. Et on écrit cet entier naturel en notant, de droite à gauche, le nombre d'objets isolés, le nombre de paquets de dix, le nombre de paquets de cent, le nombre de paquets de mille, etc. Chacun de ces nombres étant compris entre zéro et neuf, seuls dix chiffres sont nécessaires : 0, 1, 2, 3, 4, 5, 6, 7, 8 et 9. Par exemple, l'écriture 2359 exprime un entier naturel formé de 9 unités, 5 dizaines, 3 centaines et 2 milliers.

Soit :	2359	=	9 * 1 +	5 * 10 +	3 * 100 +	2 * 1000
		=	$9 * 10^{\cdots}$ +	$5 * 10^{\cdot\cdot}$ +	$3 * 10^{\cdots}$ +	$2 * 10^{\cdots}$

En *notation binaire*, c'est-à-dire en notation à position en base deux, le nombre treize s'écrit 1101 : de droite à gauche, 1 unité, 0 deuzaine, 1 quatraine et 1 huitaine.

Soit :	$(1101)_2$	=	1 * 1 +	0 * 2 +	1 * 4 +	1 * 8
		=	$1 * 2^{\cdots}$ +	$0 * 2^{\cdot\cdot}$ +	$1 * 2^{\cdots}$ +	$1 * 2^{\cdots}$

Comme la mémoire des ordinateurs est constituée de circuits qui ne peuvent être chacun que dans deux états, on peut utiliser chaque circuit de la mémoire pour représenter un chiffre binaire, en identifiant l'un de ces états avec le chiffre binaire 0 et l'autre avec le chiffre binaire 1.
Le nombre 13 = 1101 est donc représenté dans la mémoire d'un ordinateur par le mot 1101, c'est-à-dire par quatre circuits respectivement dans les états 1, 0, 1 et 1

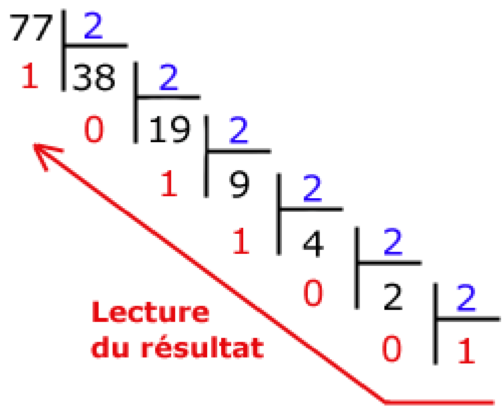
Exercice 1 : Trouver la représentation en base dix du nombre $(1011)_2$.

Exercice 2 : Quel est le plus grand nombre à 4 chiffres que l'on puisse écrire en base deux. Donner sa représentation en base deux et en base 10. Même question pou un nombre à 8 chiffes (codé su un octet)

Exercice 3 : Compléter la deuxième colonne du premier tableau de l'aide mémoire du Dauphin.

Vous savez maintenant passer de la représentation en base deux à la représentation en base 10, pour faire le contraire on peut utiliser la méthode suivante :

« Quand on a *n* objets, on les groupe par paquets de deux, qu'on regroupe eux-mêmes en paquets de deux paquets, etc. Autrement dit, on fait une succession de divisions par 2, jusqu'à obtenir un quotient égal à 0. »



$(77)_{10} = (1001101)_2$

Exercice 4 : Trouver la représentation en base deux des nombres 25 et 123.

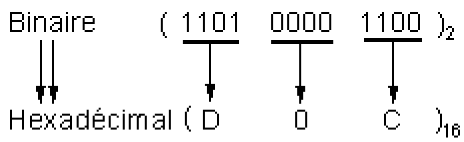
Et l'hexadécimal ?

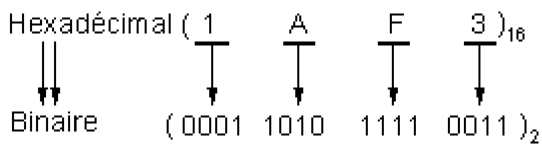
L'hexadécimal correspond à la base seize.

Pour la base dix, nous avons besoin de dix chiffres différents : 0, 1, 2, …9.
Pour la base deux, nous avons besoin de deux chiffres différents : 0 et 1.
Pour la base seize, nous avons besoin de seize chiffres différents : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ………………

La base seize est très pratique pour condenser l'écriture des nombres en base deux !

Par exemple, le nombre 135 s'écrit en base deux : $(10000111)_2$, on regroupe les chiffres par 4 en partant de la droite, et on donne la représentation en base seize de chaque nombre ainsi obtenu. Soit $(10000111)_2 = (1000\ 0111)_2 = (8\ 7)_{16}$





Exercice 5 : Vérifier que $(87)_{16}$ est bien la représentation en base seize de 135, en adaptant la méthode vue pour trouver la représentation en base dix, d'un nombre en base deux.

Exercice 6 : Donner la représentation en base seize de 183 (trouver 2 méthodes différentes)

(Texte largement inspiré du livre : Informatique et sciences du numérique – Gilles Dowek – Editeur : Eyrolle)