
# I Ecriture d'un entier dans une base b ≥ 2

Avant de s'attaquer à la base 2, ou binaire qui comme vous le savez est la représentation utilisée par les machines pour coder l'information, nous allons revenir au système décimal et au passage de la représentation d'un entier positif dans différentes bases.

## 1) Le système décimal

Il s'appuie sur 10 symboles :0,1,2,3,4,5,6,7,8,9

Prenons par exemple le nombre 678.

En fait, ce nombre a été construit de la manière suivante : $678= 6*10^2+7*10+8$

6 est le chiffre de rang 2 (ou centaine), 7 celui de rang 1 (ou dizaine) et 8 celui de rang 0 ou unités.

Tous les nombres écrits dans le système décimal sont décomposables à l'aide des puissances de 10. Lorsque le rang des unités est plein (à 9), on augmente le rang suivant d'une unité et on réinitialise à 0 le rang qui était plein. Si le rang suivant est lui aussi plein , on réinitialise à 0 les deux rangs et on ajoute 1 au rang suivant, etc…

## 2) Le système binaire

Le système binaire s'appuie sur deux symboles, le 0 et le 1 , mais fonctionne exactement comme le système décimal . Dès qu'un rang est plein, on augmente de 1 celui d'après et on réinitialise à 0. Voici la construction de l'écriture des premiers entiers en binaire.

| Nombre en décimal | Nombre en binaire |
|---|---|
| 0 | 0 |
| 1 | 1 |
| 2 | 10 |
| 3 | 11 |
| 4 | 100 |
| 5 | 101 |
| 6 | 110 |
| 7 | 111 |
| 8 | 1000 |
| 9 | 1001 |
| 10 | 1010 |

Remarque : Pour écrire le troisième symbole, comme le rang 0 est plein, on ajoute 1 au rang 1 et on réinitialise à 0 le rang 0. Pour écrire le cinquième symbole, on réinitialise les deux premiers rangs et on ajoute 1 au rang 2, etc.…

Sur les ordinateurs, le système de comptage est en binaire. En effet, un ordinateur est composé de circuits électriques et le binaire permet de matérialiser les deux états : Il y a du courant (1) ou il n'y en a pas (0). On ne parle pas de rangs, mais de bits, qui est l'unité de base.

Par exemple, le nombre en binaire 101 est codé sur 3 bits.

## 3) Passage du système décimal au binaire et réciproquement

### *a) Du décimal au binaire*

Il existe une méthode très simple pour passer du décimal au binaire. Cette méthode est fondée sur l'algorithme d''Euclide et les restes successifs de la division euclidienne par 2 de l'entier que l'on veut convertir. On obtient une succession de reste (0 ou 1) . Il suffit de les écrire du dernier obtenu , qui doit être 1, au premier.

Exemple : Convertissons 145 en binaire

| | |
|---|---|
| 145= 72*2+1<br>72=36*2+1<br>36=18*2+0<br>18=9*2+0<br>9=4*2+1<br>4=2*2+0<br>2=1*2+0<br>1=2*0+1 | La conversion de 145 en binaire est donc :<br><br>**10010011** |

### *b) du binaire au décimal*

Les nombres écrits en binaire suivent le même principe que ceux écrits en décimal mais avec les puissances de deux.

Ainsi $101_2$= $1*2^2+0*2^1+1*2^0=5$

Donc, quand on a un nombre binaire, il suffit de multiplier chaque nombre qui le compose par la puissance de 2 correspondante au rang de son bit et d'additionner tous les résultats.

Remarque :

$14= 7*2= (3*2+1)*2=((2*1+1)*2+1)*2=1*2^3+1*2^2+1*2^1+0*2^0=1110_2$.On retrouve la décomposition à l'aide des puissances de 2.

## 4) Le système hexadécimal

Le système hexadécimal s'appuie sur 16 caractères, les chiffres du système décimal plus les lettres A,B,C,D ,E,F (qui correspondent respectivement aux valeurs 10,11,12,13,14 et 15) . De la même façon que pour les deux bases vues précédemment, un nombre écrit en hexadécimal se décompose à l'aide des puissances de 16 et donc le passage de la base 16 à 10 est simple .

Ainsi $A8_{16}$= $11*16+8=184_{10}$

De même pour obtenir un nombre en hexadécimal à partir d'un nombre en décimal, on utilise la méthode des divisions successives par 16. Il suffit alors de réécrire les restes obtenus en commençant par le dernier jusqu'au premier.

| | |
|---|---|
| 98= 16*6+2<br>2=0*16+2 | 62 est donc l'écriture en hexadécimal de 98 |

## 5) Système hexadécimal et système binaire.

On l'a déjà dit, les représentations en machine se font à l'aide du système binaire et il va être plus aisé de travailler avec des multiples de 2 qu'avec le système décimal.

Le passage du binaire à l'hexadécimal se fait se façon simple en regroupant par 4 à partir de la droite les bits codant le nombre en binaire. Au besoin, on rajoute des 0 à gauche pour obtenir un multiple de 4. Il suffit alors de remplacer chaque groupe de 4 bits par sa représentation en hexadécimal.

| Binaire (base 2) | Décimal (base 10) | Hexadécimal (base 16) |
| --- | --- | --- |
| 0 | 0 | 0 |
| 1 | 1 | 1 |
| 10 | 2 | 2 |
| 11 | 3 | 3 |
| 100 | 4 | 4 |
| 101 | 5 | 5 |
| 110 | 6 | 6 |
| 111 | 7 | 7 |
| 1000 | 8 | 8 |
| 1001 | 9 | 9 |
| 1010 | 10 | A |
| 1011 | 11 | B |
| 1100 | 12 | C |
| 1101 | 13 | D |
| 1110 | 14 | E |
| 1111 | 15 | F |

## 6) Ecriture en base b quelconque

Le principe d'écriture d'un nombre écrit dans le système décimal en base b est le même :

On obtient les restes successifs de la division euclidienne du nombre par b et on recopie les restes du dernier obtenu au premier.

$n = \sum_{i=0}^{p} c_i * b^p$ où les $c_i$ sont des entiers entre 0 et b-1 .

Exemple 25= $2*3^2+2*3^1+1*3^0 = 221_3$

## 7) Compléments

On a évoqué le fait que l'information était codée à l'aide de bits. Les bits sont regroupés en bytes (ou octets ) : Il s'agit de 8 bits. L'octect, si il n'est pas l'unité minimale va servir de référence pour évaluer la place prise par les fichiers en informatique .

- 1 kilo-octet (Ko) = 1024 octets ($2^{10}$ octets)
- 1mega-octet (Mo) = 1024 kilo-octets
- 1giga-octet (Go) = 1024 mega-octets
- 1tera-octets (To)= 1024 Go.

**Attention : 1kO $\neq$ 1000 octets**

## Exercices

1) Combien peut on coder de nombres sur 1 bit ? 8 bits ? 16 bits ? n bits ?

2) Convertissez en binaire 77,255, 1023

3) Convertissez en décimal les nombres binaires suivants : 101,1111,101101010.

4) Convertissez en décimal les nombres FF, 4D1, AAA,1234 écrits en hexadécimal.

5) Convertissez en hexadécimal les nombres suivants : $458_{10}$, $6987_{10}$, $101010100110_{2}$, $11101001010011_{2}$

6) Ecrire un algorithme qui permettent de donner l'écriture en binaire d'un nombre connaissant son écriture décimale.

7) Ecrire un algorithme qui permet de connaitre l'écriture décimale d'un entier connaissant son écriture binaire .