NSI : leçon 2 : Ecriture des nombres entiers en binaire et en hexadécimal.

1. Ecriture des entiers en binaire.

A. Représentation des nombres en informatiques.

Fondamentalement tout objet numérique est représenté par des nombres.

Par construction, tout matériel informatique ne peut traiter que les nombres 0 et 1.La plus petite information que peut traiter un système informatique est donc un **bit** qui ne peut prendre que deux valeurs : 0 ou 1.

On regroupe le plus souvent les bits par groupe de 8 bits qui constitue un **octet**.
Exemple d'octet : 10100001 Le bit et l'octet sont des unités fondamentales de l'informatique. Par exemple, l'adresse IP v4 d'un matériel informatique sur le réseau Internet est définie sur 32 bits. Pour faciliter la lecture d'une telle adresse, on la décompose en _ octets.

Exercice 1

a) Donner un exemple d'adresse IP v4.

b) Déterminer, au million près, le nombre d'adresses différentes IP v4 que permet ce protocole.

Ne pouvant traiter que les chiffres 0 et 1, un ordinateur calcule donc en **binaire** (c'est à dire en **base 2**).

Ce système d'écriture des nombres en base 2 est antérieur aux ordinateurs, puiqu'il a été proposé par le mathématicien Gottfried Wilhelm Leibniz :

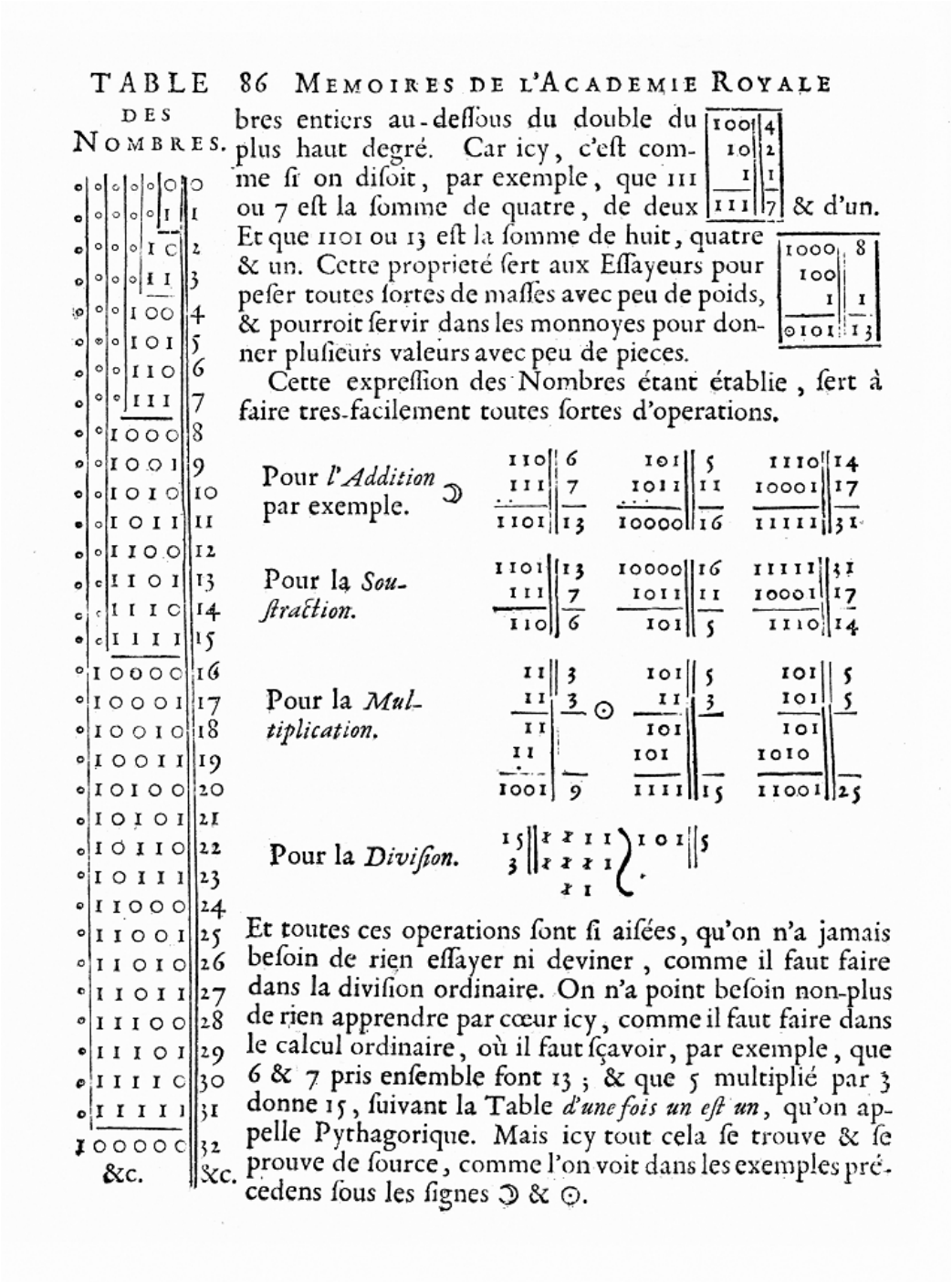


TABLE DES NOMBRES.

86 MEMOIRES DE L'ACADEMIE ROYALE

0 0 0 0 0	0
0 0 0 0 1	1
0 0 0 1 0	2
0 0 0 1 1	3
0 0 1 0 0	4
0 0 1 0 1	5
0 0 1 1 0	6
0 0 1 1 1	7
0 1 0 0 0	8
0 1 0 0 1	9
0 1 0 1 0	10
0 1 0 1 1	11
0 1 1 0 0	12
0 1 1 0 1	13
0 1 1 1 0	14
0 1 1 1 1	15
1 0 0 0 0	16
1 0 0 0 1	17
1 0 0 1 0	18
1 0 0 1 1	19
1 0 1 0 0	20
1 0 1 0 1	21
1 0 1 1 0	22
1 0 1 1 1	23
1 1 0 0 0	24
1 1 0 0 1	25
1 1 0 1 0	26
1 1 0 1 1	27
1 1 1 0 0	28
1 1 1 0 1	29
1 1 1 1 0	30
1 1 1 1 1	31
1 0 0 0 0 0	32
&c.	&c.

bres entiers au-dessous du double du plus haut degré. Car icy, c'est comme si on disoit, par exemple, que 111 ou 7 est la somme de quatre, de deux & d'un. Et que 1101 ou 13 est la somme de huit, quatre & un. Cette proprieté sert aux Essayeurs pour peser toutes sortes de masses avec peu de poids, & pourroit servir dans les monnoyes pour donner plusieurs valeurs avec peu de pieces.

Cette expression des Nombres étant établie, sert à faire tres-facilement toutes sortes d'operations.

Pour *l'Addition* par exemple. ☽

Pour la *Soustraction.*

Pour la *Multiplication.* ⊙

Pour la *Division.*

Et toutes ces operations sont si aisées, qu'on n'a jamais besoin de rien essayer ni deviner, comme il faut faire dans la division ordinaire. On n'a point besoin non-plus de rien apprendre par cœur icy, comme il faut faire dans le calcul ordinaire, où il faut sçavoir, par exemple, que 6 & 7 pris ensemble font 13 ; & que 5 multiplié par 3 donne 15, suivant la Table *d'une fois un est un*, qu'on appelle Pythagorique. Mais icy tout cela se trouve & se prouve de source, comme l'on voit dans les exemples précedens sous les signes ☽ & ⊙.

Page de l'article : "Explication de l'Arithmétique Binaire", 1703/1705.

Ce système a été développé par Liebniz alors qu'il effectuait des recherches pour mettre au point une machine mécanique de calculs.

B. Passage du binaire au décimal.

Un octet étant la représentation en base 2 d'un nombre, il est facile de le traduire en décimal :

$2^7 = 128$	$2^6 = 64$	$2^5 = 32$	$2^4 = 16$	$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$	**en decimal**
0	1	1	0	1	1	0	1	= 64 + 32 + 8 + 4 + 1 = 109
1	0	0	0	0	1	0	1	=
1	0	1	0	0	0	0	1	=
0	0	1	1	1	1	1	1	=

Exercice 2

Pour faciliter la lecture d'une adresse IP v4, on la représente sous la forme de 4 nombres entiers séparés par un point.

Exemple d'adresse IP V4 : 125.10.56.3

Pour obtenir cette représentation, on décompose l'écriture de l'adresse sur 32 bits en 4 octects que l'on traduit en décimal et que l'on sépare par un point.

1) Quelle adresse IP V4 représente le nombre suivant ?

01000101010101010110010001001101

2) 312.55.78.200 peut-il représenter une adresse IP v4 ?

3) Quelle est l'adresse IP V4 la plus grande que l'on puisse écrire ?

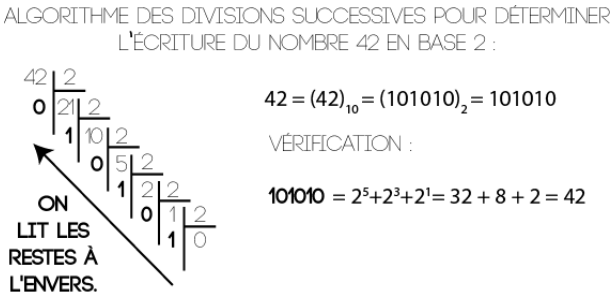
C. Passage du décimal au binaire.

Pour traduire un nombre écrit en base 10 en base 2, deux méthodes sont possibles.

a) La décomposition en puissances de 2.

Par exemple : $15 = 8 + 4 + 1 = 2^3 + 2^2 + 2^0 = 1101$

b) La méthode des divisions successives.



Exercice 3

a) Traduire en binaire, par la méthodes de votre choix, les nombres suivantes :

- 12 =

- 150 =
- $2^{10} + 2^5$ =

b) Combien faut-il de bits pour écrire le nombre 2019 en binaire ?

> En python, pour définir qu'un nombre est écrit en binaire, il suffit de le faire précéder par Øb (zéro suivi de b) :
>
> ```
> >>>øb01101101
> 109
> ```

D. D'autres systèmes de représentation des nombres en informatique.

L'écriture des nombres en binaire étant particulièrement longue et pas toujours facile à déchiffrer, on a développé en informatique d'autres formes de représentation des nombres.

a) Un vieux système d'écriture des nombres que l'on retrouve parfois.

Sur le forum https://forum.ubuntu-fr.org/, un utilisateur a demandé de l'aide : un programme scientifique concernant les noyaux donnent en sortie :

```
0330440 125117 106321 037021 130245 144067 036305 036775 107762
0330460 061145 017166 037007 161500 010550 141466 037023 162313
0330500 144557 054711 037037 032162 033270 143327 037046 065167
0330520 142067 137052 037056 114047 031200 105575 037063 020435
0330540 013773 120241 037067 054032 013711 044537 037073 067762
0330560 166312 030467 037076 140571 110177 010327 037100 104351
0330600 032173 104542 037100 104353 032173 104542 037100 140576
0330620 110177 010327 037100 070000 166312 030467 037076 054050
0330640 013711 044537 037073 020447 013773 120241 037067 114057
0330660 031200 105575 037063 065177 142067 137052 037056 032167
```

source : https://forum.ubuntu-fr.org/viewtopic.php?id=2032984

Exercice 4

a) A votre avis en quelle base sont écrit ces différents nombres ?

b) La première colonne représente des adresses obtenues par additions successives d'un même nombre. Lequel ?

> À partir de la version 3 de Python, un nombre octal commence par Øo (zéro suivi de la lettre minuscule o) ou ØO (zéro suivi de la lettre capitale O)
>
> ```
> >>> Øo20
> 16
> ```

---

b) Un système toujours utilisé et à connaître.

Suite à une conversion par un logiciel, les nombres représentés dans les colonnes 2 à 9, ont changé de représentation :

```
0330340 8e30 967d 3e4d e6fb 2f71 e5c7 3e47 0850
0330360 5a8f 93a2 3e42 3dc0 bb85 e435 3e3b 56ca
0330400 d443 294c 3e34 2174 96be df35 3e2b 247d
0330420 bfc5 6850 3e22 0ccd 1625 03a9 3e18 4b50
0330440 aa4f 8cd1 3e11 b0a5 c837 3cc5 3dfd 8ff2
0330460 6265 1e76 3e07 e340 1168 c336 3e13 e4cb
0330500 c96f 59c9 3e1f 3472 36b8 c6d7 3e26 6a77
0330520 c437 be2a 3e2e 9827 3280 8b7d 3e33 211d
0330540 17fb a0a1 3e37 581a 17c9 495f 3e3b 6ff2
0330560 ecca 3137 3e3e c179 907f 10d7 3e40 88e9
```

Exercice 5

a) A votre avis en quelle base sont écrit les nombres dans les colonnes 2 à 9 ?

b) Pensez-vous que ces deux tableaux représentent les mêmes données ?

> En Python, un nombre hexadécimal s'écrit en le commençant par les deux caractères « Øx" (zéro suivi de x) comme suit :
>
> ```
> >>> Øx33
> 51
> ```

E. Résumé des commandes python

Résumé des écritures particulières des nombres en Python

```
# spécifications des écritures des nombres en python v3
>>> 0x18 # écriture en héxadécimal
24
>>> 0o30 # écriture en octal
24
>>> 0b11000 # écriture en binaire
24
```

Commandes bien utiles

```
# Quelques commandes bien utiles
>>> hex(25) # conversion en héxadécimal
'0x19'
>>> bin(12) # conversion en binaire
'0b1100'
>>> oct(16) # conversion en octal
'0o20
```