Séquence 1: Encodage

I. Histoire de l'informatique

La notion d'informatique est née il y a fort longtemps. Il y a eu beaucoup d'évènement qui nous ont menés, petit à petit, à l'informatique telle que nous la connaissons aujourd'hui. En voici, dans un premier temps un résumé :

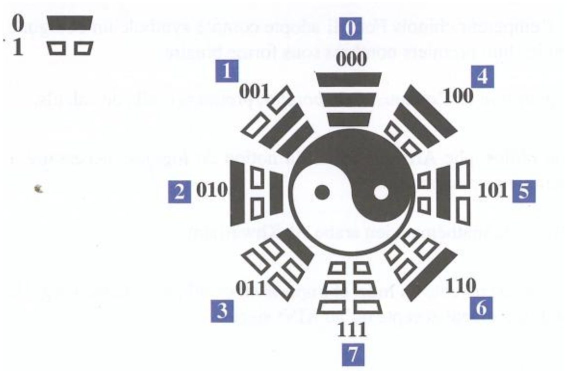
- -3000 : l'empereur chinois Fou-Hi adopte comme symbole un octogone contenant les huit premiers nombres sous forme binaire.
- -500 : apparition de l'*abaque* et du *boulier*, premiers outils de calculs.
- -300 : le philosophe Aristote définit la notion de *logique*, nécessaire à l'informatique contemporaine.
- 820 : travaux du mathématicien Al-Khwârizmî.
- 1000 : le zéro est inventé en Inde, et importé en occident au Moyen-âge. Il ne sera définitivement accepté qu'au XIVe siècle.
- 1600 : l'écossais John Napier invente les logarithmes, qui permettent de ramener les multiplications et les divisions à des additions et des soustractions.

  A la même époque, l'Allemand Wilhelm Zchickard invente ce qu'il appele une horloge calculante, pouvant réaliser des additions, soustractions, multiplications et mémorisations des résultats.
- 1697 : introduction du binaire en Europe par le mathématicien Gottfried Leibniz.
- 1840 : Augusta Lovelace pose le principe d'une machine à calculer. Trois années plus tard, elle définit la théorie de la programmation.
- 1854 : George Boole définit la théorie de la logique binaire.

Développons certains points de cette chronologie.

1. L'octogone de l'empereur Fou-Hi

La notation binaire est plus ancienne que l'on pourrait l'imaginer … En – 3000, l'empereur chinois Fou-Hi crée un symbole magique que l'on peut considérer comme la première expression d'un codage en binaire, sous forme d'un octogone comportant huit *trigrammes*. Chaque trigramme est une association de trois signes qui peuvent être soit une bande pleine noire, soit une bande creuse coupée en deux.



Si une bande noire représente 0 et une bande creuse coupée représente un 1, on peut compter de 0 à 7 en binaire avec les huit trigrammes, ainsi que le montre la figure suivante.

2. Aristote et la logique

La logique aristotélicienne sera essentielle pour clarifier tout raisonnement déductif, et l'informatique nécessite de tels raisonnements.
Aristote introduit les syllogismes, raisonnement logiques mettant en relation trois propositions. Les deux premières sont appelées prémisses et la dernière, obtenue à partir des deux précédentes, est appelées conclusion.

Définition 1 : syllogisme

Un syllogisme est un raisonnement où deux prémisses permettent d'émettre une conclusion.

Exemples :

- « Tous les hommes sont mortels, or Socrate est un homme donc Socrate est mortel ». Ce syllogisme est dû à Aristote lui-même.
  Ici la première prémisse est : « tous les hommes sont mortels » ; c'est un fait avéré.
  La seconde prémisse est : « Socrate est un homme » ; c'est aussi un fait avéré qui n'a pas besoin d'être prouvé.
  La conclusion est : « Socrate est mortel » ; des deux faits avérés précédemment cités, on peut conclure ceci.

- Toute planète de notre système solaire possède un rayon supérieur à 2000 kilomètres. Mercure est une planète de notre système solaire. Donc Mercure a un rayon supérieur à 2000 km.

ATTENTION :
Il existe des syllogismes menant à une conclusion déroutante. Le plus connu est le suivant :
« Tout ce qui est rare est cher. Or, un cheval bon marché est rare. Donc un cheval bon marché est cher. »
On appelle cela un *sophisme*.

3. Al-Khwârizmî

Al-Khwârizmî, de son vrai nom *Muhamad Ibn Mūsā al-Khuwārizmī* était un savant perse, né dans les années 780. Son nom latinisé en *Algoritmi* est à l'origine du mot *algorithme*.
Il est l'auteur d'un ouvrage mathématique, *Kitabu al-muskhtasar fī hisābi al-jabr waal-muqābalah*, considéré comme le premier ouvrage d'algèbre.
Dans ce titre, on distingue la partie « al-jabr », qui désigne une opération, et qui est à l'origine du mot « algèbre ».

4. Invention du zéro

Il est sûrement surprenant de savoir que le chiffre «0 » fut inventé après les chiffres 1, 2, etc., mais c'est le cas.

Avant son invention, les indiens n'avaient pas l'utilité de ce nombre (le zéro en tant que chiffre existait déjà depuis l'époque mésopotamienne, mais n'avait pas d'utilité dans les calculs).

C'est le mathématicien et astronome indien Brahmagupta qui l'introduisit en tant que nombre dans son ouvrage *Brâhma Siddhânta*. Le mot fut traduit ensuite en arabe par le mot « sifr », qui désignait le vide, qui donna plus tard le mot « chiffre ».

Le mot « zéro », quant à lui, vient de l'italien zephiro, traduction de sifr dans cette langue.

Le zéro fut ensuite introduit en Europe par le mathématicien italien Leonardo Fibonacci, et simplifia considérablement les calculs (jusque là faits à l'aide d'abaques, donc peu pratiques).

Quant au symbole «0 », il semblerait provenir de la représentation (initialement un cercle) de la voûte céleste.

5. Le binaire de Leibniz

Gottfried Leibniz, personnage allemand aux multiples talents (mathématiques, philosophie, diplomatie, juriste, philologie, …), était passionné par la Dyadique, c'est-à-dire par tout ce qui se rapporte à la réunion de deux principes philosophiques qui se complètent réciproquement.

S'interrogeant sur l'utilité d'un système connu des chinois, qui peut se comparer à un système binaire, il expose ses idées à l'Académie des sciences de Paris, et celles-ci furent publiées dans un ouvrage intitulé Explication de l'arithmétique binaire avec des remarques sur son utilité et sur le sens qu'elle donne des anciennes figures chinoises de Fou-Hi.

6. Augusta «ada » Lovelace

Augusta Lovelace était la fille du poète romantique Lord George Byron et d'une mathématicienne.

Elle est à l'origine du « Principe des machines à calculer ». Un langage de programmation porte son prénom en mémoire de ses travaux.

Pour elle, une machine à calculer doit comporter :

- un dispositif permettant d'introduire les données numériques (cartes perforées, roues dentées…)
- une mémoire pour conserver les valeurs numériques entrées
- une unité de commande grâce à laquelle l'utilisateur va indiquer à la machine les tâches à effectuer
- un « moulin » chargé d'effectuer les calculs
- un dispositif permettant de restituer les résultats (imprimante, écran…)

Ces principes seront, un siècle plus tard, à la base des premiers ordinateurs.

Elle définit ensuite le principe d'itérations successives dans l'exécution d'une opération. En l'honneur du mathématicien arabe Al-Khwârizmi, elle appelle «algorithme » le processus logique permettant l'exécution d'un programme.

7. L'algèbre de Boole

Dans son ouvrage *Les lois de la pensée*, George Boole explique que l'on peut coder les démarches de la pensée à l'aide de systèmes n'ayant que deux états : ZERO-UN ; OUI-NON ; VRAI-FAUX, etc. L'algèbre de Boole, ou calcul booléen, voit alors le jour.

Il existe deux types d'opérations dans l'algèbre de Boole.

Propriété 1 : conjonction logique

Soit B un ensemble fini de deux éléments (B = {0 ; 1} ou B = {faux ; vrai} par exemple). La fonction « ET » est définie sur B par le tableau suivant :

ET	0	1
0	0	0
1	0	1

équivalent à

ET	faux	vrai
faux	faux	faux
vrai	faux	vrai

Propriété 2 : disjonction Logique

Soit B un ensemble fini de deux éléments (B = {0 ; 1} ou B = {faux ; vrai} par exemple). La fonction « OÙ » est définie sur B par le tableau suivant :

OU	0	1
0	0	1
1	1	1

équivalent à

OU	faux	vrai
faux	faux	vrai
vrai	vrai	vrai

Il existe d'autres propriétés sur cette algèbre, mais ce n'est pas l'objet de ce cours.

Les booléens sont très utiles en programmation ; nous aurons l'occasion de nous en apercevoir ultérieurement dans ce livre.

8. Le début du XX° siècle

C'est en 1936 qu'apparaît le concept de machine universelle, capable d'exécuter tous les algorithmes, et que les notions de machine, algorithme, langage et information sont pensées comme un tout cohérent. Les premiers ordinateurs ont été construits en 1948 et leur puissance a ensuite évolué exponentiellement, c'est-à- dire avec une croissance très rapide.

Un ordinateur ne peut stocker que des nombres. Pour écrire du texte, on associe à chaque caractère un entier naturel et la correspondance entre le caractère et son code est appelé un *charset*. La façon de transcrire un texte grâce aux codes des caractères qui le composent est appelé *Encoding* (encodage).

II. Les encodages

1. Le binaire

Définition 2 : bit

> Le bit (contraction de Binary Digit) est l'unité la plus simple en informatique. Il ne peut prendre que deux valeurs (désignées le plus souvent par 0 et 1).

Un bit sert essentiellement à représenter un « état » : vrai ou faux, oui ou non, le courant électrique passe ou ne passe pas, etc.

Remarque : le mot « bit » est certes une contraction, mais il y a aussi un jeu de mot avec le mot anglais « bit » qui signifie « petit morceau ».
Le mot «bit» a été popularisé par Claude Shannon, ingénieur en génie électrique et mathématicien américain du XXe siècle, qui en attribue l'invention à John Tukey, l'un des plus importants mathématiciens américain du XX° siècle.

Définition 3 : système binaire

> Le système binaire est le système de numération utilisant comme base le bit.

Le système binaire permet d'écrire ou d'encoder des nombres ou des caractères.

2. L'ASCII

Les premiers textes étaient encodés à l'aide d'une sorte d'alphabet, un tableau contenant 128 caractères codés en binaire, c'est-à-dire 2^8 caractères, afin que l'on puisse encoder les caractères avec seulement 7 bits. En effet, les ordinateurs utilisaient des cases mémoires de un octet, mais ils réservaient toujours le 8e bit pour le contrôle de parité (c'est une sécurité pour éviter les erreurs, qui étaient très fréquentes dans les premières mémoires électroniques).

Ce tableau contenait les caractères les plus utilisés dans la langue anglaise : les lettres de l'alphabet en majuscule (de A à Z) et en minuscule (de a à z), les dix chiffres arabes (de O à 9), des signes de ponctuation (point, virgule, point- virgule, deux points, points d'exclamation et d'interrogation, apostrophe ou quote, guillemets ou double quote, parenthèses, crochets etc.), quelques symboles et certains caractères spéciaux invisibles (espace, retour-chariot, tabulation, retour- arrière, etc.)

ASCII est l'acronyme d'*American Standard Code for Information Interchange.*

Code décimal	Caractère
0	[NUL]
...	...
27	[ESC]

...	...
33	!
34	"
35	#
...	...

Code décimal	Caractère
48	0
49	1
...	...
57	9
...	...
65	A
66	B
...	...
90	Z
91	[
...	...
97	a
98	b
...	...
122	z
123	{
...	...
127	[DEL]

Extrait de la table ASCII

Exemple : le caractère « A » est codé en ASCII par le nombre 65 (dans le système décimal), qui correspond au nombre binaire 1000001, et au nombre hexadécimal 41.

Définition 4 : octet

Un octet est un octuplet (un multiplet de 8 bits), c'est-à-dire une suite de 8 bits.

Exemple : « 10000001 » est l'octet représentant le caractère « A ».

Ainsi, chaque caractère d'un texte est codé sur un octet.

Un texte de 10 000 caractères est donc codé sur 10 x 1 000 octets, soit 10 kilo- octets (10 ko).

L'ASCII a tout de même un inconvénient majeur : celui de ne pouvoir coder les caractères accentués, qui apparaissent dans plusieurs langues, dont le français.

Il a donc fallu trouver un moyen d'étendre la table ASCII...

3. naissance de l'ISO

Le problème de la norme ISO 8859-1, dont le nom complet est ISO/CEI 8859-1 et qui est souvent appelée Latin-1 ou Europe occidentale, fit son apparition dans le but de remédier à l'inconvénient de l'ASCII. Elle forme la première partie de la norme internationale ISO/CEI 8859, qui est une norme de l'*Organisation internationale de normalisation* pour le codage des caractères en informatique.

En France, depuis l'apparition de l'euro, c'est le codage ISO-8859-15 (sauvent appelé Latin-9 qui permet d'écrire tout ce que l'on veut dans notre langue. C'est une sorte de mise à jour de la norme ISO 8859-1.

4. Au niveau mondial : l'Unicode

Le problème de la norme ISO 8859-15 se trouve principalement dans les échanges de textes à l'échelle mondiale.

Imaginez que vous codiez un programme en France et que vous souhaitiez collaborer avec un collègue Indien ou Russe. Ce dernier n'utilise pas nécessairement le même encodage que vous et va donc avoir quelques problèmes quand il lira vos propres codes.

Il est donc primordial, dans ce cas de figure, d'utiliser un encodage qui puisse être utilisé dans tous les pays. C'est le but d'Unicode, dont la première publication à octobre 1991.

Définition 5 : point de code

Un point de code est la valeur numérique qui représente un caractère dans l'espace de stockage des caractères.

Unicode accepte plusieurs formes de transformations universelles pour représenter un point de code valide. Citons par exemple :

- UTF-8
- UTF-16
- UTF-32

Le nombre après UTF représente le nombre minimal de bits de codets (c'est-à-dire des groupes de signes représentant une information) avec lesquels un point de code valide est représenté.

L'UTF-8 est par exemple le plus commun pour les applications comme Unix et Internet. **Son codage de taille variable (voir exercice5)** lui permet d'être en moyenne moins coûteux en occupation mémoire.

L'UTF-16 est un bon compromis lorsque la place mémoire n'est pas trop restreinte, car la grande majorité des caractères Unicode assignés pour les écritures des langues modernes (dont les caractères les plus fréquemment utilisés) peuvent être représentés sur 16 bits.

L'UTF-32 est utilisé lorsque la place mémoire n'est pas un problème et que l'on a besoin d'avoir accès à des caractères de manière directe et sans changement de taille (par exemple les hiéroglyphes égyptiens). L'avantage de cette transformation normalisée est que tous les codets ont la même taille. Il n'est donc pas nécessaire de lire des codets supplémentaires pour déterminer le début de la représentation d'un point de code. Toutefois, ce format est particulièrement peu économique (y compris en mémoire) puisqu'il « gaspille » inutilement au moins un octet (toujours nul) par caractère.

5. Les logiciels

Selon le système d'exploitation que l'on utilise, il existe quelques éditeurs de textes qui permettent de choisir l'encodage des caractères.

Par exemple,

- sous Windows : Notepad ++ ; Sublimetext
- sous OSX : TextEdit ;
- sous Unix (Linux par exemple) : Kwrité ou Geany.

III. Les bases

Définition 6 : intervalle d'entiers

Soient m et n deux nombres entiers. On note $[\![m:n]\!]$l'ensemble de tous les entiers compris entre m (inclus) et n (inclus).

1. Principe général

Soit N un entier naturel.

Si : $N = a_n \times b^n + a_{n-1} \times b^{n-1} + \dots + a_1 \times b^1 + a_0 \times b^0$, $a_k \in [\![0:a-1]\!]$,

Alors on dira que l'écriture de n en base b est :

$$N = \overline{a_n.a_{n-1}..a_1a_0}^b$$

Remarque : afin de savoir en quelle base on écrit un nombre, on l'écrit en le surmontant d'un barre suivie de la base.

Exemple : soit N tel que

$$N = 3 \times 5^4 + 4 \times 5^3 + 1 \times 5^1 + 2 \times 5^0,$$

Alors,

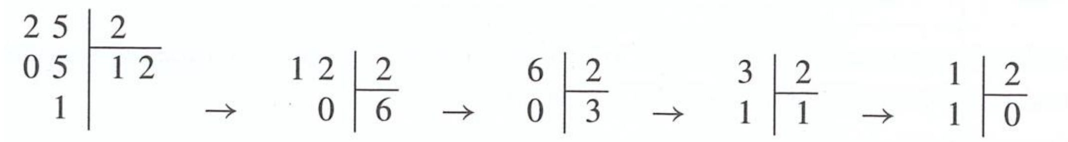
$$N = \overline{34012}^b$$

2. En base 2

La base 2 est le système binaire. Elle est donc très importante en informatique. La base est constituée uniquement de deux nombres : 0 et 1. Ainsi, tout nombre écrit en base 2 ne comportera que des 0 et des 1.

Pour convertir un nombre décimal en base 2, il faut effectuer des divisions euclidiennes successives par 2 et écrire les restes obtenus en « remontant ».

Exemple : on souhaite convertir 25 en base 2.



On en déduit que :

$25 = 1 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 + 1 \times 2^4$

soit :

$25 = \boxed{1} \times 2^0 + \boxed{0} \times 2^1 + \boxed{0} \times 2^2 + \boxed{1} \times 2^3 + \boxed{1} \times 2^4$

Ainsi, $\overline{25}^{10} = \overline{11001}^{2}$.   On peut aussi noter le nombre b11001 (avec b pour binaire)

MÉTHODE Après avoir posé toutes les divisions, on écrit tous les restes en partant du dernier obtenu jusqu'au premier.

3. La base hexadécimale

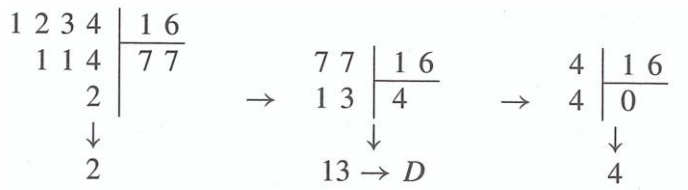
Elle est constituée de 16 caractères :

0 ; 1 ; 2 ; 3 ; 4 ; 5 ; 6 ; 7 ; 8 ; 9 ; A ; B ; C ; D ; E ; F.

3.1 - Du décimal à l'hexadécimal

Pour convertir un nombre décimal en hexadécimal, on effectue des divisions euclidiennes par 16, sur le même principe que l'écriture en base 2.

Exemple : on souhaite convertir le nombre décimal 1 234 en hexadécimal.



Ainsi,

$\overline{1234}^{10} = \overline{4D2}^{16}$ que l'on peut aussi noter x4D2 ou h4D2 (avec x ou h pour hexadécimal)

3.2 - Du binaire à l'hexadécimal

Pour convertir du binaire à l'hexadécimal, il suffit de regrouper les bits par groupe de 4 en allant de la droite à la gauche.

Exemple : on souhaite convertir $\overline{01001101}^{2}$ en hexadécimal.

Binaire	0100	1101
Pseudo-décimal	4	13
Hexadécimal	4	D

résultat:

$$\overline{01001101}^{2} = \overline{4D}^{16}$$

3.3 - <u>De l'hexadécimal au décimal</u>

Le principe est le même que pour passer du binaire au décimal, si ce n'est que l'on va remplacer les lettres par les nombres correspondants (on remplace A par 10, B par 11....).

Exemple : on souhaite convertir $\overline{7EF}^{16}$ en décimal.

- F correspondant au nombre 15, est au rang 0 ;
- E, correspondant au nombre 14, est au rang 1 ;
- 7 est au rang 2.

On doit donc calculer :

$$7 \times 16^{2} + 14 \times 16^{1} + 15 \times 16^{0} = 2\,031.$$

Ainsi,

$$\overline{7EF}^{16} = \overline{2031}^{10}$$

Exercices types :

1)
$\overline{1000001}^{2}$ correspond à la lettre « A ». Pour obtenir le binaire correspondant à la lettre « B », il faut lui ajouter 1.
Il y a deux façons de faire :
- soit on convertit 66 en binaire (car 65 est le code ASCII de « A », donc 66 est celui de «B »),
- soit on prend le suivant de 1000001 . Ce binaire se terminant par 1, son suivant se termine par 0, et il faut ajouter 1 au bit précédent.
On trouve alors que $\overline{1000010}^{2}$ est le binaire correspondant à «B ».

2)
Pour convertir $\overline{75}^{10}$ en binaire, il faut regarder les divisions euclidiennes par 2 :
On prend ensuite les restes dans l'ordre inverse dans lequel nous les avons trouvés. On obtient alors que $\overline{75}^{10}$ = $\overline{1001011}^{2}$

3)
Pour convertir $\overline{2001}^{10}$ en hexadécimal, il faut diviser par 16
On prend alors les restes dans l'ordre inverse dans lequel nous les avons trouvés : il y en a un qui dépasse 9. Comme 10 correspond à A, 13 correspond donc à D. On obtient que $\overline{2001}^{10}$ = $\overline{7D1}^{16}$

4)

Pour convertir ABC en binaire, on peut commencer par le convertir en décimal.

- A, correspondant au nombre 10, occupe le rang 2, cette colonne « vaut » 16^2;
- B, correspondant au nombre 11, occupe le rang 1, cette colonne « vaut » 16 ;
- C, correspondant au nombre 12, occupe le rang 0, cette colonne « vaut » 1 ;

On doit donc calculer :

$10 \times 16^2 + 11 \times 16^1 + 10 \times 16^0 = 2748$.

Ainsi, $\overline{ABC}^{16}$ = $\overline{2748}^{10}$

Pour convertir $\overline{2748}^{10}$ en binaire, afin d'éviter les divisions euclidiennes successives (ce qui peut paraître long), on peut se baser sur les puissances successives de 2 :

n	2^n
0	1
1	2
2	4

3	8
4	16
5	32
n	2^n
6	64
7	128
8	256
9	512
10	1024
11	2048

Les coefficients des puissances de 2 sont donc répartis ainsi :

$$\begin{aligned}2748 &= 2048 + 512 + 128 + 32 + 16 + 8 + 4 \\ &= 2^{11} + 2^9 + 2^7 + 2^5 + 2^4 + 2^3 + 2^2.\end{aligned}$$

On a donc :

2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	1	0	1	0	1	1	1	1	0	0

Ainsi, $\overline{ABC}^{16} = \overline{2748}^{10} = \overline{101010111100}^{2}$