Les nombres

- La première utilisation des nombres a été de compter, de dénombrer
  - Entiers naturels (entiers non signés)
- Il a fallu ensuite exprimer la notion de gain ou de perte
  - Entiers relatifs (entiers signés)
- Ensuite se sont posés des problèmes de partage et de mesure
  - Nombres réels (nombres en virgule flottante)

Représentation des nombres

- Un nombre est une entité abstraite
- On ne peut le manipuler qu'à travers une représentation
- Une représentation d'un nombre est tributaire d'un support de dimension finie
- L'ensemble des nombres représentables est donc fini

Représentation des entiers naturels

- La représentation la plus naturelle est celle des jeux de cartes ou de dominos
  - Elle consiste à représenter un nombre *N* par un ensemble d'éléments identiques dont le cardinal est *N*
  - Elle devient vite inexploitable quand la valeur de *N* augmente
- On a donc défini des systèmes plus condensés
  - Système juxtapositionnel des romains
  - Système de numération positionnelle

Numération positionnelle

- Le principe consiste à se donner un ensemble de *B* symboles qui représentent les entiers de 0 à *B*-1
- *B*, cardinal de l'ensemble, est appelé **base de numération**
- On utilise les symboles suivants, appelés **chiffres**
  - *B* = 2, *A* = {0, 1}
  - *B* = 10, *A* = {0, 1, ..., 8, 9}
  - *B* = 16, *A* = {0, 1, ..., 8, 9, A, B, C, D, E, F}

Développement de base *B* d'un nombre *N*

Pour représenter un entier naturel non nul *N* dans une base *B*, on part d'un ensemble dont le cardinal est *N*.

On regroupe les éléments par paquets de *B*. Il reste alors un nombre a_0 d'éléments tel que $a_0 < B$.

Les paquets de *B* éléments peuvent être regroupés en paquets de *B* paquets. Il reste alors un nombre a_1 de paquets tel que $a_1 < B$.

En continuant de la sorte et en notant B^2 les paquets de paquets, B^3 les paquets de paquets de paquets, etc..., il existe un $a_{p-1} \neq 0$ tel que :

$$N = a_{p-1} B^{p-1} + a_{p-2} B^{p-2} + \cdots + a_1 B + a_0 = \sum_{i=0}^{p-1} a_i B^i$$

$$avec \; \forall i \in [0, \ldots, p-1] \quad a_i < B$$

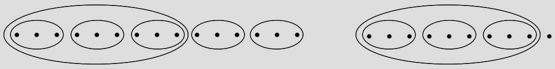
Un tel développement existe toujours et est unique.

Exemples en base 10

$N_1 = 1\times10 + 5$
$= 1\times B^1 + 5\times B^0$

$N_2 = 1\times10 + 0$
$= 1\times B^1 + 0\times B^0$

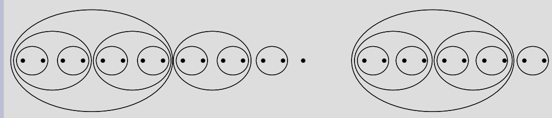
Exemples en base 3



$N_1 = 1\times9 + 2\times3 + 0$
$= 1\times B^2 + 2\times B^1 + 0\times B^0$

$N_2 = 1\times9 + 0\times3 + 1$
$= 1\times B^2 + 0\times B^1 + 1\times B^0$

Exemples en base 2



$N_1 = 1\times8 + 1\times4 + 1\times2 + 1$
$= 1\times B^3 + 1\times B^2 + 1\times B^1 + 1\times B^0$

$N_2 = 1\times8 + 0\times4 + 1\times2 + 0$
$= 1\times B^3 + 0\times B^2 + 1\times B^1 + 0\times B^0$

Représentation par un langage de codification

- On va représenter les entiers naturels à l'aide d'un langage de codification
- Ce langage peut être :
  - de format variable à structure morphologique restrictive
    - c'est le format que nous utilisons dans les calculs manuels
  - de format fixe à structure morphologique non restrictive
    - c'est le format utilisé en machine

Langage à format variable

- Soit L_B^{*n} un langage défini par
  - Alphabet $A = \{0, 1, ..., B\text{-}1\}$ de base B,
  - Format variable $\leq n$,
  - Règle de restriction : on interdit les mots de format > 1 commençant par le chiffre 0.
- On peut montrer que sa puissance lexicographique est B^n
- Ce langage permet donc de représenter les entiers appartenant à $\mathbb{N}/B^n$

Représentation d'un naturel en format variable

On commence par associer à un naturel *N* son développement de base *B* qui est un polynôme de degré *p-1* < *n* dont la variable est *B* et les coefficients sont des naturels inférieurs à *B*. Soit P_N ce polynôme :

$$P_N = \sum_{i=0}^{p-1} a_i B^i$$

La représentation de *N*, notée *N**, est constituée des coefficients de P_N écrits dans l'ordre des degrés décroissants :

$$N^* = a_{p-1}a_{p-2}...a_1a_0$$

Exemples en format variable

- Base 10

  $P_{N_1} = 1\cdot B + 5 \qquad N_1^* = 15$

  $P_{N_2} = 1\cdot B + 0 \qquad N_2^* = 10$

- Base 3

  $P_{N_1} = 1\cdot B^2 + 2\cdot B + 0 \qquad N_1^* = 120$

  $P_{N_2} = 1\cdot B^2 + 0\cdot B + 1 \qquad N_2^* = 101$

- Base 2

  $P_{N_1} = 1\cdot B^3 + 1\cdot B^2 + 1\cdot B + 1 \qquad N_1^* = 1111$

  $P_{N_2} = 1\cdot B^3 + 0\cdot B^2 + 1\cdot B + 0 \qquad N_2^* = 1010$

Langage à format fixe

- Soit L_B^n un langage défini par
  - Alphabet A = {0, 1, ..., B-1} de base B,
  - Format fixe n,
  - Structure morphologique non restrictive.
- Par définition, sa puissance lexicographique est B^n
- Ce langage permet donc de représenter les entiers appartenant à $\mathbb{N}/B^n$

Représentation d'un naturel en format fixe

Comme pour la représentation en format variable, on associe au naturel N, le polynôme P_N correspondant à son développement de base B :

$$P_N = \sum_{i=0}^{p-1} a_i B^i$$

La représentation de N, notée N^{*n}, est constituée des coefficients de P_N écrits dans l'ordre des degrés décroissants précédés de n-p chiffres 0 :

$$N^{*n} = a_{n-1} \ldots a_p a_{p-1} a_{p-2} \ldots a_1 a_0$$

avec $\forall i \in \{p, ..., n\text{-}1\}$ $a_i = 0$

Exemples en format fixe

- Base 10, format 4

$$P_{N_1} = 1 \cdot B + 5 \qquad N_1^{*4} = 0015$$
$$P_{N_2} = 1 \cdot B + 0 \qquad N_2^{*4} = 0010$$

- Base 3, format 4

$$P_{N_1} = 1 \cdot B^2 + 2 \cdot B + 0 \qquad N_1^{*4} = 0120$$
$$P_{N_2} = 1 \cdot B^2 + 0 \cdot B + 1 \qquad N_2^{*4} = 0101$$

- Base 2, format 4

$$P_{N_1} = 1 \cdot B^3 + 1 \cdot B^2 + 1 \cdot B + 1 \qquad N_1^{*4} = 1111$$
$$P_{N_2} = 1 \cdot B^3 + 0 \cdot B^2 + 1 \cdot B + 0 \qquad N_2^{*4} = 1010$$

Représentation machine des naturels

- Les bases les plus couramment utilisées sont les base 2 (binaire) et 10 (décimal)
- L'utilisation de la base 10 nécessite une représentation des chiffres décimaux dans un langage binaire
- On distinguera 2 types de représentation
  - Binaire pur (base 2)
  - Décimal codé binaire (base 10)

Naturels en binaire pur

- À tout naturel $N \in \mathbb{N}/2^n$ on fait correspondre le polynôme de son développement et une représentation symbolique $N^{*n} \in L_2^n$
- Exemples

$$n = 4, |L_2^4| = 16, N = douze$$
$$P_N = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$$
$$N^{*4} = 1100$$

$$n = 8, |L_2^8| = 256, N = trente\ sept$$
$$P_N = 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$
$$N^{*8} = 00100101$$

Naturels en décimal codé binaire

- À tout naturel $N \in \mathbb{N}/10^n$ on fait correspondre le polynôme de son développement et une représentation symbolique $N_n^* \in L_{10}^n$
- Chaque chiffre décimal doit ensuite être codé dans un langage binaire
- Il y a 10 chiffres décimaux, le format des mots doit donc être égal à 4 ($2^3 < 10 \leq 2^4$)
- On utilise donc un langage L_2^4 qui contient 16 mots
- On va définir 2 codes

Code BCD

- Ce code consiste à convertir la valeur du chiffre décimal en binaire sur 4 positions
- C'est un code pondéré 8-4-2-1
- La table de correspondance est la suivante

Chiffre	*Code*	*Chiffre*	*Code*
0	0000	5	0101
1	0001	6	0110
2	0010	7	0111
3	0011	8	1000
4	0100	9	1001

Code Excess 3

- Ce code s'obtient en ajoutant 3 à chaque valeur du code BCD
- C'est un code autocomplémenteur à 9
- La table de correspondance est la suivante

Chiffre	*Code*	*Chiffre*	*Code*
0	0011	5	1000
1	0100	6	1001
2	0101	7	1010
3	0110	8	1011
4	0111	9	1100

Exemple en décimal codé binaire

- Soit le naturel N = *trois cent douze* à représenter en décimal codé binaire sur 4 positions

$$(N^{*4})_{10} = 0312$$

$$(N^{*4})_{BCD} = 0000\ 0011\ 0001\ 0010$$

$$(N^{*4})_{Excess3} = 0011\ 0110\ 0100\ 0101$$

Représentation des entiers relatifs

- On veut représenter les entiers de $\mathbb{N}/B^n$ ainsi que leurs opposés
- Pour cela, on définit l'ensemble suivant
  $\mathbb{Z}/B^n = \{N \in \mathbb{Z} \mid |N| \in \mathbb{N}/B^n\}$
  $= \{-(B^n-1), \ldots, -1, 0, 1, \ldots, B^n-1\}$
- Attention, cette notation, $\mathbb{Z}/B^n$, est propre à ce cours !!!
- $|\mathbb{Z}/B^n| = 2B^n-1$

Représentation des entiers relatifs

- Pour coder cet ensemble dans un langage binaire de format fixe, il faut des mots de longueur $n+1$
- En effet, en rappelant que $B \geq 2$ et $n \geq 1$
  - n ne suffit pas car $B^n < 2B^n-1$
  - $n+1$ suffit car $B^{n+1} \geq 2B^n > 2B^n-1$

$$B^{n+1} \geq 2B^n$$
$$B^{n+1} - 2B^n \geq 0$$
$$B^n(B-2) \geq 0$$

Représentation des entiers relatifs

- On étudiera 2 codages
  - Représentation en valeur absolue et signe
    - C'est la représentation usuelle des relatifs
  - Représentation en complément vrai
    - Elle possède des propriétés intéressante pour certaines opérations
- Pour ces 2 codages, la représentation d'un entier positif est identique à celle du naturel correspondant, avec une position de plus

Représentation en valeur absolue et signe (VA+S)

- On réserve la position de tête pour le signe et les n autres pour la valeur absolue
  - Le signe + est représenté par le chiffre 0
  - Le signe - est représenté par le chiffre B-1
- On a donc

$$N^{*(n+1)} = a_n a_{n-1} \ldots a_1 a_0$$
$$avec$$
$$a_n = 0 \; si \; N \geq 0 \; et \; a_n = B-1 \; si \; N < 0$$
$$|N|^{*n} = a_{n-1} \ldots a_1 a_0$$

Représentation en complément vrai (CV)

- Un relatif N qui appartient à $\mathbb{Z}/B^n$ est représenté par un naturel sur n+1 positions
- Si $N \geq 0$ $\quad N^{*(n+1)} = |N|^{*(n+1)}$
- Si $N < 0$ $\quad N^{*(n+1)} = (B^{n+1} - |N|)^{*(n+1)}$
- Il résulte de cette définition que la représentation d'un positif commence par un 0 et celle d'un négatif par un B-1

Représentation machine

- Comme pour les naturels, on n'utilise que le binaire pur (base 2) et le décimal codé binaire (base 10)
- Les machines actuelles représentent toutes les relatifs binaires en complément vrai
  - Les opérations d'addition et de soustraction sont plus faciles
  - Par contre, les multiplications et les divisions, moins fréquentes, sont plus difficiles à faire

Relatifs en binaire pur

- À chaque relatif N qui appartient à $\mathbb{Z}/2^n$ est associé un mot du langage L_2^{n+1}

$$(N^{*(n+1)})_2 = a_n a_{n-1} \ldots a_1 a_0$$

- Exemples de représentation
  - VA+S

$$(-1^{*4})_2 = 1001 \qquad (+5^{*4})_2 = 0101$$

  - CV

$$(-1^{*4})_2 = 1111 \qquad (+5^{*4})_2 = 0101$$

Relatifs en décimal codé binaire

- À chaque relatif N qui appartient à $\mathbb{Z}/10^n$ est associé un mot du langage L_{10}^{n+1}

$$(N^{*(n+1)})_{10} = a_n a_{n-1} \ldots a_1 a_0$$

- Exemples de représentation
  - VA+S

$$(-1^{*2})_{10} = 91 \qquad (+5^{*2})_{10} = 05$$
$$(-1^{*2})_{BCD} = 1001\ 0001 \qquad (+5^{*2})_{BCD} = 0000\ 0101$$
$$(-1^{*2})_{Excess3} = 1100\ 0100 \qquad (+5^{*2})_{Excess3} = 0011\ 1000$$

  - CV

$$(-1^{*2})_{10} = 99 \qquad (+5^{*2})_{10} = 05$$
$$(-1^{*2})_{BCD} = 1001\ 1001 \qquad (+5^{*2})_{BCD} = 0000\ 0101$$
$$(-1^{*2})_{Excess3} = 1100\ 1100 \qquad (+5^{*2})_{Excess3} = 0011\ 1000$$

Représentation des réels

- La représentation exacte d'un nombre réel est souvent infinie
- Un nombre réel est donc approché par un nombre appelé **décimal** si la base est 10 ou ***B*-naire** pour une base B quelconque

Les nombres *B*-naires

- On appelle *B*-naire tout nombre rationnel dont un représentant a un dénominateur qui est une puissance entière de la base *B*
- *X* est *B*-naire si et seulement si

$$\exists N \in \mathbb{Z}, \exists d \in \mathbb{N}, X = \frac{N}{B^d}$$

- Il y a une infinité de représentants possibles
- Il y en a un qui correspond au *d* le plus petit, c'est le représentant *B*-naire le plus simple

Développement généralisé de base *B*

- Soit le représentant *B*-naire le plus simple de $|X| = \frac{|N|}{B^d}$
- Soit le développement du numérateur

$$|N| = a_{p-1} B^{p-1} + a_{p-2} B^{p-2} + \cdots + a_1 B + a_0 \quad avec \quad a_{p-1} \neq 0$$

- Alors $|X| = \frac{|N|}{B^d}$ a un développement qui dépend de *d*.
- Il faut envisager 3 cas

Développement généralisé de base *B* (cas 1)

- Si 0 < *d* ≤ *p*-1, le développement est

$$|X| = a_{p-1} B^{p-1-d} + \cdots + a_d B^0 + a_{d-1} B^{-1} + \cdots + a_0 B^{-d}$$

$$|X| = \sum_{i=0}^{p-1} a_i B^{i-d} \quad avec \quad a_{p-1} \neq 0 \; et \; a_0 \neq 0$$

- Il se décompose en deux parties
  - La partie entière qui possède *p*-*d* termes

$$a_{p-1} B^{p-1-d} + \cdots + a_{d+1} B + a_d$$

  - La partie fractionnaire qui possède *d* termes

$$a_{d-1} B^{-1} + a_{d-2} B^{-2} + \cdots + a_0 B^{-d}$$

Développement généralisé de base *B* (cas 2)

- Si *d* = 0, le développement ne comprends que la partie entière car

$$d = 0 \Rightarrow |X| \in \mathbb{N} \quad et \quad |X| = |N|$$

- Le développement est donc celui de |*N*|, il ne comprends que la partie entière

$$|X| = a_{p-1} B^{p-1} + a_{p-2} B^{p-2} + \cdots + a_1 B + a_0$$

$$|X| = \sum_{i=0}^{p-1} a_i B^i \quad avec \quad a_{p-1} \neq 0$$

Développement généralisé de base *B* (cas 3)

- Si *p*-1 < *d*, il n'existe pas de terme de rang *d* dans le développement de |*N*|, on ajoute donc *d*-*p*+1 termes nuls

$$|N| = 0 \cdot B^d + 0 \cdot B^{d-1} + \cdots + 0 \cdot B^p + a_{p-1} B^{p-1} + \cdots + a_1 B + a_0$$

$$|X| = 0 \cdot B^0 + 0 \cdot B^{-1} + \cdots + 0 \cdot B^{p-d} + a_{p-1} B^{p-1-d} + \cdots + a_0 B^{-d}$$

$$|X| = \sum_{i=0}^{d} a_i B^{i-d} \quad avec \quad a_{p-1} \neq 0 \; et \; \forall \, j \in \{p, \ldots, d\}, a_j = 0$$

  La partie entière a un seul terme de coefficient nul, la partie fractionnaire comprends *d* termes dont les *d*-*p* premiers ont des coefficients nuls

Représentation en virgule fixe

- On représente un *B*-naire comme un entier sur lequel on place une virgule entre les positions *d* et *d*-1 (sauf dans le cas 2)
  - Si le développement est (cas 1)

$$|X| = a_{p-1} B^{p-1-d} + \cdots + a_d B^0 + a_{d-1} B^{-1} + \cdots + a_0 B^{-d}$$

  - La représentation correspondante est

$$|X|^* = a_{p-1} \cdots a_d , a_{d-1} \cdots a_0$$

  - Le cas 3 est similaire

- Cette représentation pose des problèmes

Problèmes en virgule fixe

- Il y a 2 problèmes
  - On ne peut pas représenter des nombres très grands ou très petits
    - La plus grande valeur absolue représentable sur *n* positions est $|X_{max}| = \frac{|N_{max}|}{B^d} = \frac{B^n - 1}{B^d}$
    - La plus petite non nulle est $|X_{min}| = \frac{|N_{min}|}{B^d} = \frac{1}{B^d}$
  - Les opérations de multiplication et de division déplacent la position de la virgule
- D'où la définition d'une autre représentation, en virgule flottante

Représentation en virgule flottante

- Soit un *B*-naire $X = \frac{N}{B^d}$
- On peut écrire $X = \frac{N}{B^d B^e} B^e$
- Le terme $M = \frac{N}{B^d B^e} = \frac{X}{B^e}$ est appelé la mantisse
  - *M* correspond à *X* dans lequel la virgule a été déplacée
    - de *e* positions à gauche si *e* ≥ 0
    - de *e* positions à droite si *e* < 0
- On a donc $X = M \cdot B^e$

Virgule flottante normalisée

- Il y a une infinité de couples (*M*,*e*) qui respectent la relation $X = M \cdot B^e$
- On définit la représentation en virgule flottante normalisée d'un nombre *B*-naire non nul par le couple
  - dont la mantisse est purement fractionnaire (la partie entière de |*M*| est nulle)
  - et dont le premier chiffre après la virgule est non nul

Représentation machine des flottants

- C'est une représentation en format fixe
- Les bases utilisées sont le binaire et l'hexadécimal
- Il faut représenter la mantisse et l'exposant
- Mantisse
  - On ne représente que la partie fractionnaire
  - Elle peut se représenter en VA+S ou en CV
- Exposant
  - C'est un entier relatif
  - Il peut se représenter en VA+S, en CV ou avec la méthode de l'exposant translaté.

Exposant translaté

- L'exposant est représenté sur *q* positions par un entier naturel *e*'

$$e' = e + K \quad avec \quad K = \frac{B^q}{2} - 1$$

- Intérêt de la méthode
  - Respecte l'ordre lexicographique
  - Dans les bases différentes de 2, permet de représenter plus d'exposants

Norme IEEE 754

- Un nombre est constitué de 3 champs
  - Le signe de la mantisse
  - L'exposant translaté
  - La valeur absolue de la mantisse
- Il existe plusieurs formats dont :
  - Simple précision sur 32 bits : 1 + 8 + 23
    - Exposant translaté de 127
  - Double précision sur 64 bits : 1 + 11 + 52
    - Exposant translaté de 1023
- Le premier bit de la mantisse normalisée n'est pas représenté