1. INTRODUCTION : SYSTÈMES DE NUMÉRATIONS

- Le système décimal est un système de numération fondé sur la position des chiffres.
  - Tout nombre est représenté à l'aide d'un nombre limité de symboles.
  - Simplicité des algorithmes de calcul.
  - Valeur dépendant de la position occupée.

Les systèmes courants :

Système	Base	Symboles
Décimal	Dix	0,1,2,3,4,5,6,7,8,9
Binaire	Deux	0,1
Octal	Huit	0,1,2,3,4,5,6,7
Hexadécimal	Seize	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

Si nous pouvons compter sur nos dix doigts, l'ordinateur lui ne peut compter que sur deux états.
Dans un ordinateur, le support physique de l'information est un élément ne présentant que deux états magnétiques stables (symbolisés par **0** et **1**).
Conséquence : Toute information (numérique ou non), doit être codée en séquence de bits, c'est-à-dire en nombre binaires.
Les deux symboles (**0** et **1**), du système binaire sont appelés bits (binary digit).

Exemple : En base 10

- Nombre de symboles : 10
- Symboles : 0,1,2,3,4,5,6,7,8,9
- $3745902 = 3 \cdot 10^6 + 7 \cdot 10^5 + 4 \cdot 10^4 + 5 \cdot 10^3 + 9 \cdot 10^2 + 0 \cdot 10^1 + 2 \cdot 10^0$
- Algorithme de calcul : $\begin{array}{r} 14 \\ +\ 6 \\ \hline 20 \end{array}$

<u>Contre-exemple : Les chiffres romains</u>

- La position ?? IV et VI
- Algorithme de calcul :

```
  XIV
+ VI
-----
  XX
```

???

Exemple d'une autre base : la base 3

Base 10	Base 3
0	0
1	1
2	2
3	10
4	11
5	12
6	20
7	21
8	22
9	100
10	101
11	102
12	110

Remarques :

- A b entiers composés d'un seul chiffre en base b
- A b^2 entiers composés d'au plus 2 chiffres en base b
- A b^n entiers composés d'au plus n chiffres en base b

Ce sont en base 10 les entiers

$$0 ;1 ;2 ;3 ;4 ;\dots ; b^n-1$$

Alors $(b^n)_{10}$ est le plus petit nombre dont l'écriture nécessite $n+1$ chiffres en base b

Exemple :

- $(3^1)_{10} = (10)_3$
- $(3^2)_{10} = (100)_3$

On en conclut que :

$$\boxed{\left(b^n\right)_{10} = \left(1\underbrace{000...00}_{\text{n chiffres}}\right)_b}$$

<u>Soit</u> $x = (\beta_i)_b$ l'écriture du nombre en base b,

Alors
$$x = \sum_{i=0}^{\infty} \beta_i \cdot \left(1\underbrace{0'0000...00}_{\text{i zéros}}\right)_b$$
$$= \sum_{i=0}^{\infty} \beta_i \cdot \left(b^i\right)_{10} \text{ (écriture du nombre en base 10)}$$

Exemple :

$$\begin{aligned} x &= (112)_3 = 1 \cdot (100)_3 + 1 \cdot (10)_3 + 2 \cdot (1)_3 \\ &= 1 \cdot 3^2 + 1 \cdot 3^1 + 2 \cdot 3^0 \\ &= 9 + 3 + 2 \\ &= (14)_{10} \end{aligned}$$

Transformation : base b $\longrightarrow$ base10

Exemple : base 3 $\longrightarrow$ base 10

$(120112)_3 \to (?)_{10}$

$$\begin{aligned}(120112)_3 &= 1\cdot 3^5 + 2\cdot 3^4 + 0\cdot 3^3 + 1\cdot 3^2 + 1\cdot 3^1 + 2\cdot 3^0 \text{ (notation étendue)}\\ &= 243 + 2\cdot 81 + 0 + 1\cdot 9 + 1\cdot 3 + 2\cdot 1\\ &= 243 + 162 + 9 + 3 + 2\\ &= (419)_{10}\end{aligned}$$

Donc $(120112)_3 = (419)_{10}$

Transformation : base 10 $\longrightarrow$ base b

Exemple : base 10 $\longrightarrow$ base 3

Méthode 1

Il faut trouver sa notation étendue (somme de puissances de 3)

$$x = (59)_{10} = (?)_3$$

$3^3 < x < 3^4$	$59 - 3^3 = 59 - 27 = 32$
$3^3 < 32 < 3^4$	$32 - 3^3 = 32 - 27 = 5$
	$59 = 3^3 + 3^3 + 5$
$3 < 5 < 3^2$	$5 - 3^1 = 2$
	$59 = 3^3 + 3^3 + 3^1 + 2$
	$59 = 2 \cdot 3^3 + 0 \cdot 3^2 + 1 \cdot 3^1 + 2 \cdot 3^0$

$$x = (59)_{10} = (2012)_3$$

Méthode 2

$$x = (59)_{10} = (?)_3$$

Division	Quotient	Reste
59 :3	19	2
19 :3	6	1
6:3	2	0
2:3	0	2

On lit les restes de bas en haut

$$x = (59)_{10} = (2012)_3$$

Remarques :

Tout nombre entier positif en base b peut s'écrire sous la forme :

$$x = (\alpha_n ; \alpha_{n-1} ; \ldots ; \alpha_2 ; \alpha_1 ; \alpha_0)_b$$
avec $\alpha_i \in \{0;1;2;3;\ldots\ldots;b-1\}$

ou en notation étendue avec $b > 1$

$$x = \alpha_n \cdot b^n + \alpha_{n-1} \cdot b^{n-1} + \ldots + \alpha_2 \cdot b^2 + \alpha_1 \cdot b^1 + \alpha_0$$
$$= \sum_{i=0}^{\infty} \alpha_i \cdot b^i \quad ,\text{avec } 0 \leq \alpha_i < b$$

Exemple :

$$(419)_{10} = 4 \cdot 10^2 + 1 \cdot 10^1 + 9$$

$$(120112)_3 = 1 \cdot 3^5 + 2 \cdot 3^4 + 0 \cdot 3^3 + 1 \cdot 3^2 + 1 \cdot 3^1 + 2$$

Transformation : base b $\longrightarrow$ base b^n

Partager le nombre en tranche de n chiffres, en partant de la droite, et remplacer chaque tranche par son équivalent en base b^n.

Exemple : base 5 $\longrightarrow$ base 125 (5^3)

$$(10120334210)_5 = (?)_{125}$$

$$(10120334210)_5 = (10|120|334|210|)_5$$
$$= (5;35;94;55)_{125}$$

Car

$$(210)_5 = 2 \cdot 5^2 + 1 \cdot 5^1 + 0 \cdot 5^0 = 55$$
$$(334)_5 = 3 \cdot 5^2 + 3 \cdot 5^1 + 4 \cdot 5^0 = 94$$
$$(120)_5 = 1 \cdot 5^2 + 2 \cdot 5^1 + 0 \cdot 5^0 = 35$$
$$(10)_5 = 1 \cdot 5^1 + 0 \cdot 5^0 = 5$$

Transformation : base b^n ⟶ base b

Remplacer chaque chiffre du nombre par son équivalent en base b (écriture avec n chiffres)

<u>Exemple :</u> base $81(3^4)$ ⟶ base 3

$$(12;6;46)_{81} = ((0110)_3;(0020)_3;(1201)_3)_{81}$$
$$= (11000201201)$$

Car

Division	Quotient	Reste
12 :3	4	0
4 :3	1	1
1 :3	0	1

Division	Quotient	Reste
6 :3	2	0
2 :3	0	2

Division	Quotient	Reste
46 :3	15	1
15 :3	5	0
5 :3	1	2
1 :3	0	1

Tout nombre réel positif en base b peut s'écrire sous la forme :

$$x = (\alpha_n ; \alpha_{n-1} ; ... ; \alpha_2 ; \alpha_1 ; \alpha_0 \ , \alpha_{-1} ; \alpha_{-2} ; \alpha_{-3} ; ...)_b$$

avec $\alpha_i \in \{0;1;2;3;.....;b-1\}$

ou en notation étendue avec $b > 1$

$$x = \alpha_n \cdot b^n + \alpha_{n-1} \cdot b^{n-1} + ... + \alpha_2 \cdot b^2 + \alpha_1 \cdot b^1 + \alpha_0$$
$$+ \alpha_{-1} \cdot b^{-1} + \alpha_{-2} \cdot b^{-2} + \alpha_{-3} \cdot b^{-3} + ...$$
$$= \sum_{i \in Z} \alpha_i \cdot b^i$$

<u>Exemple :</u>

$$(419,52)_{10} = 4 \cdot 10^2 + 1 \cdot 10^1 + 9 \cdot 10^0 + 5 \cdot 10^{-1} + 2 \cdot 10^{-2}$$

$$(120,112)_3 = 1 \cdot 3^2 + 2 \cdot 3^1 + 0 \cdot 3^0 + 1 \cdot 3^{-1} + 1 \cdot 3^{-2} + 2 \cdot 3^{-3}$$

Transformation : base b $\longrightarrow$ base10

Ecrire le nombre en notation étendue.
(somme de puissance de b)

Exemple :

$$(1001{,}1011)_2 = 1\cdot 2^3 + 0\cdot 2^2 + 0\cdot 2^1 + 1\cdot 2^0 + 1\cdot 2^{-1} + 0\cdot 2^{-2} + 1\cdot 2^{-3} + 1\cdot 2^{-4}$$
$$= 9{,}6875$$

Transformation : base10 $\longrightarrow$ base b

On traite séparément la partie entière et la partie décimale du nombre.
La partie entière : idem qu'un entier positif.
La partie décimale :
Multiplier cette partie décimale et les parties décimales des produits successifs par b.
La séquence des parties entières des produits obtenus constitue la représentation en base b de la partie décimale du nombre.

Exemple :

$(11{,}4368)_{10} = (?)_5$

1. $(11)_{10} = (21)_5$
2. $(0{,}4368)_{10} = (0{,}2043)_5$

	PE	PD
$0{,}4368\cdot 5 =$	2	0,184
$0{,}184\cdot 5 =$	0	0,92
$0{,}92\cdot 5 =$	4	0,6
$0{,}6\cdot 5 =$	3	0

Nombres à développement infini périodique

1. $(0,\bar{1})_b = \dfrac{1}{b-1}$
2. $(0,\bar{\beta})_b = \dfrac{\beta}{b-1}$

<u>Preuve de 1 :</u>

Posons $x = (0,\bar{1})_b = (0{,}1111111111...)$

Alors $b \cdot x = (10)_b \cdot (0,\bar{1})_b = (1,\bar{1})_b$

$= 1_b + (0,\bar{1})_b = 1 + x$

D'où $bx - x = 1$

$x(b-1) = 1 \qquad x = \dfrac{1}{b-1}$

On conclut que $(0,\bar{1})_b = \dfrac{1}{b-1}$

<u>Preuve de 2 :</u>

$$(0,\bar{\beta})_b = \beta \cdot (0,\bar{1})_b = \beta \cdot \frac{1}{b-1} = \frac{\beta}{b-1}$$

Transformation : base b^n $\longleftrightarrow$ base b

La transformation des réels positifs utilise le même algorithme que les entiers positifs

Exemple 1 :

$$(0{,}10221)_3 = (\ldots)_9$$

$$\begin{aligned}(0{,}10221)_3 &= (0{,}10|22|10|)_3 \\ &= (0{,}383)_9\end{aligned}$$

Exemple 2 :

$$((15),(5;23))_{27} = (\ldots)_3$$

$$\begin{aligned}((15),(5;23))_{27} &= ((120)_3,((012)_3;(212)_3)) \\ &= (120{,}012212)_3\end{aligned}$$

LES SYSTEMES BINAIRE, OCTAL ET HEXADECMAL

Avantage du système décimal : compacité de l'écriture. Pour le système binaire, les chaînes de bits sont difficiles à distinguer.

Pour se rapprocher du système décimal, en partant du système binaire (qui est utilisé pour la représentation de l'information dans un ordinateur), les systèmes octal (base 8) et hexadécimal (base 16) sont alors utilisés.

La raison pour laquelle ce sont ces deux bases qui ont été choisies, est la suivante

ce sont les puissances de 2 qui encadrent 10 : $2^3 < 10 < 2^4$

Pour les conversions binaire-octal et binaire-hexadécimal, on procédera selon les méthodes vues

Conversion d'un chiffre octal en sa représentation binaire à 3 bits :

Chiffre octal	Equivalent binaire
0	000
1	001
2	010
3	011
4	100
5	101
6	110

Conversion d'un chiffre hexadécimal en sa représentation binaire à 4 bits:

Chiffre hexadécimal	Equivalent décimal	Equivalent binaire
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

LES COMPLÉMENTS

Exemple 1 :

$$843 - 236 = ?$$

$$\begin{aligned} 843 - 236 &= 843 + [1000 - 236] - 1000 \\ &= 843 + \left[(999 - 236) + 1\right] - 1000 \\ &= 843 + \underbrace{763}_{\text{complément à 9 de 236}} + 1 - 1000 \\ &= 843 + \underbrace{764}_{\text{complément de 236}} - 1000 \\ &= 1607 - 1000 \\ &= 607 \end{aligned}$$

Donc soustraire c'est additionner le complément

A = 843 B =236 Y=A-B = ? A>B

$B : 236$

complément à 9 de $B : 763$

complément de $B : 764$

$$\begin{array}{r} 843 \\ +\,764 \\ \hline 1607 \\ \hline 607 \end{array}$$

Exemple 2 :

A = 27 B =152 Y=A-B = ? A<B

$$
\begin{array}{ll}
B: 152 & 27 \\
\text{complément} & +848 \\
\text{à 9 de } B: 847 & \overline{875} \\
\text{complément} & \\
\text{de } B: 848 &
\end{array}
$$

Mais 27-152 ≠875, mais 27-152 = -125.
Par contre 875 est le complément de la valeur absolue de la solution.

$$
\begin{array}{lr}
 & 27 \\
 & +848 \\
\text{complément } Z: & \overline{875} \\
\text{complément à 9 de } Z: & 874 \\
Z: & 125
\end{array}
$$

Conclusion :

A-B=Y si A > B alors (A et B sur 4 digits)

$$\boxed{A - B = A + (\text{complément de } B) - 1000 = Y}$$

A-B=Y si A < B alors (A et B sur 4 digits)

$$\boxed{A - B = A + (\text{complément de } B) - 1000 = -(\text{complément de } Y)}$$

Exemple 3 : (Complément binaire)

$A = (1001)_2$ $B = (101)_2$ $Y = A - B = ?$ $A > B$

$$\begin{array}{ll} B : 0101 & \quad 1001 \\ \text{complément} & \underline{+1011} \\ \text{à 1 de} \quad B : 1010 & \quad 10100 \\ \text{complément} & \quad\;\; 100 \\ \text{de} \quad B : 1011 & \end{array}$$

$$y = (100)_2$$

Exemple 4 : (Complément binaire)

$A = (101)_2$ $B = (1001)_2$ $Y = A - B = ?$ $A < B$

$$\begin{array}{lll} B : 1001 & 101 & \\ \text{complément} & \underline{+111} & \\ \text{à 1 de} \quad B : 0110 & 1100 & \text{complément} \\ \text{complément} & 1011 & \text{complément à 1} \\ \text{de} \quad B : 0111 & 0100 & |Y| \end{array}$$

$$y = -(100)_2$$

LA REPRÉSENTATION DES ENTIERS NÉGATIFS

Les compléments peuvent être utilisés pour donner une représentation des entiers négatifs

Exemple : $(121)_4 \rightarrow (121)_4$; $(121)_4 \rightarrow (-213)_4$

Conséquence :

1) Sens du premier digit
2) Fixer le nombre de positions

Exemple : • En base 4

entiers négatifs (commencent par 2 ou 3)

entiers positifs (commencent par 1 ou 0)

• Sur 4 positions

Entiers positifs de $(0000)_4 \longrightarrow 0_{10}$

à $(1333)_4 \longrightarrow 2 \cdot 4^3 - 1 = 127_{10}$

Entiers négatifs de

$$\begin{array}{r} (2000)_4 \\ - \quad 1 \\ \hline (1333)_4 \end{array}$$

(2000) valeur absolue $= -2 \cdot 4^3 = -128_{10}$

à

$$\begin{array}{r} (3333)_4 \\ - \quad 1 \\ \hline (3332)_4 \end{array}$$

(0001) valeur absolue $= -1 = -1_{10}$

Le code binaire direct

Dans ce paragraphe, on envisagera uniquement la représentation des nombres entiers à l'intérieur d'un ordinateur.

Le code binaire direct nécessite que les nombres soient enregistrés dans l'ordinateur sous la forme d'un nombre fixé de bits.

Représentation des nombres entiers :

un entier est représenté en mémoire par son expression binaire s'il est positif ou nul, et par le complément de sa valeur absolue s'il est négatif

Pour la suite de ce paragraphe, on supposera qu'un nombre entier est représenté à l'aide d'un mot de8bits (=1 octet) :

Exemple 1

Considérons le nombre $A = 57$

Conversion en binaire

Représentation de A :

$A = 57 = 32+ 16+8+ 1 = (111001)_2$

0	0	1	1	1	0	0	1

Représentation de 57

Représentation de -A

complément à 1 de A complément de A

A :	0 0 1 1 1 0 0 1
complément à 1 de A :	1 1 0 0 0 1 1 0
complément de A :	1 1 0 0 0 1 1 1

1	1	0	0	0	1	1	1

Représentation de - 57

<u>Remarque 1</u>

Pour reconnaître si un nombre entier est positif ou négatif, il suffit de tester le premier bit

si ce premier bit est 0 , alors le nombre est positif si c'est un 1 , alors le nombre est négatif

Quels sont alors les entiers qui peuvent être représentés ?

entiers positifs :

0	0	0	0	0	0	0	0

qui représente 0

0	1	1	1	1	1	1	1

qui représente 127 [MAXINT]

entiers négatifs :

1	0	0	0	0	0	0	0

qui représente –128 [MININT]

1	1	1	1	1	1	1	1

qui représente -1

Liste des entiers représentables à l'aide du code binaire direct avec un mot de 8 bits

-128 ; -127 ; … ; -1 ; 0 ; 1; …… ;127

Remarque 2 :

Au lieu du code binaire direct, on peut envisager le codage suivant le premier bit est réservé au signe du nombre : 0 si + , 1 si - ;
les sept autres bits sont réservés pour la valeur absolue du nombre.

Quels sont alors les entiers qui peuvent être représentés ?

entiers positifs :

0	0	0	0	0	0	0	0

qui représente 0

0	1	1	1	1	1	1	1

qui représente 127

entiers négatifs :

1	0	0	0	0	0	0	0

qui représente – 0

1	1	1	1	1	1	1	1

qui représente -127

Un inconvénient le 0 est codé de deux manières différentes.

Exemple 2

[Utilisation pour les calculs]

Calculons 85-57 = ? A=85 , B=57

A - B = 85 + (-57) = A + (-B)

	0	1	0	1	0	1	0	1	A
+	1	1	0	0	0	1	1	1	-B
1	0	0	0	1	1	1	0	0	

Le 1 en entête qui doit être supprimé, est perdu automatiquement

Exemple 3

Calculons Y = B-A = 57-85

	0	0	1	1	1	0	0	1	B
+	1	0	1	0	1	0	1	1	-A
	1	1	1	0	0	1	0	0	

Il n'y a pas de 1 en entête à supprimer, car A > B

Le premier bit est un 1, donc le résultat représente un entier négatif :

11100100 est le complément de la valeur absolue de Y

$$\begin{array}{r} 11100100 \\ -1 \\ \hline 11100011 \end{array}$$

11100011 est le complément à 1 de la valeur absolue de Y

00011100 est la valeur absolue de Y

Donc : $Y = -(11100)_2 = -\left(2^4 + 2^3 + 2^2\right) = -28$

<u>Remarque</u>

Le code binaire direct permet de faire sans problème une addition ou une soustraction d'entiers à l'aide de la seule opération d'addition.