

LIVRET D'EXERCICES : REPRÉSENTATION DE NOMBRES ENTIERS (correction)

Exercice 1

Compléter le tableau

Base	Nom	Symboles utilisés
2	binaire	0,1
8	octale	0 à 7
10	décimale	0 à 9
16	hexadécimale	0 à 9 puis A à F

Exercice 2 : écriture des nombres

a. 34_{10} en base 2

$$34 = 2 \times 17$$

$$34 = 2 \times (2 \times 8 + 1)$$

$$34 = 2 \times (2 \times 2 \times 2 + 1)$$

$$34 = 2 \times (2^4 + 1)$$

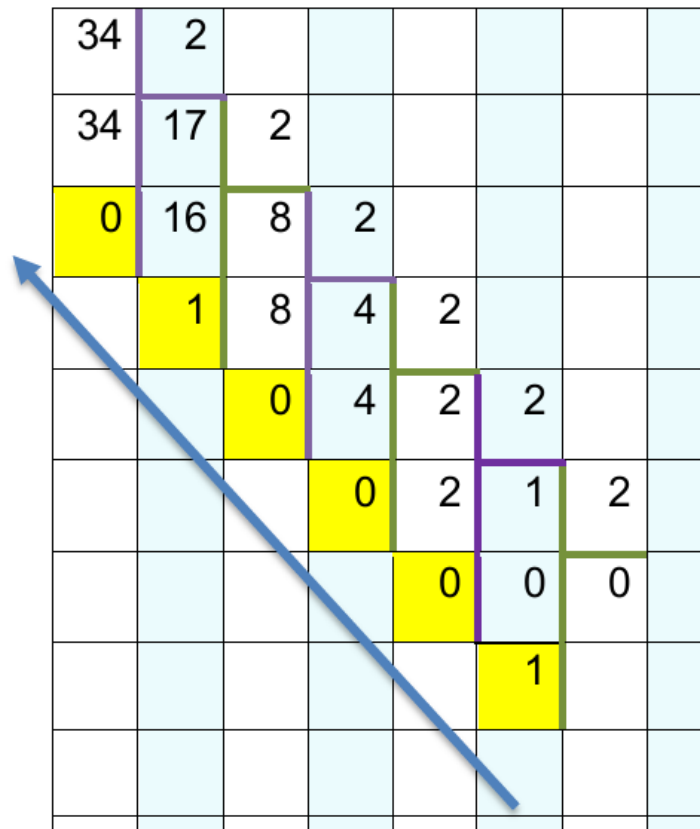
$$34 = 2 \times 2^4 + 2 \times 1$$

$$34 = 2^5 + 2^1$$

Séquence	1	0	0	0	1	0
Position	5	4	3	2	1	0
Poids	2^5	2^4	2^3	2^2	2^1	2^0

$$34_{10} = 100010_2$$

Ou bien :



b. 101011_2 en base 10

Séquence	1	0	1	0	1	1
Position	5	4	3	2	1	0
Poids	2^5	2^4	2^3	2^2	2^1	2^0

$$101011_2 = (1 \times 2^5 + 1 \times 2^3 + 1 \times 2^1 + 1 \times 2^0)_{10}$$

$$101011_2 = (32 + 8 + 2 + 1)_{10}$$

$$101011_2 = (43)_{10}$$

c. 1111111_2 en base 10

Séquence	1	1	1	1	1	1	1	1
Position	7	6	5	4	3	2	1	0
Poids	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

$$1111111_2 = (1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0)_{10}$$

$$1111111_2 = (128 + 64 + 32 + 16 + 8 + 4 + 2 + 1)_{10}$$

$$1111111_2 = 255_{10}$$

d. 57_{10} en base 2

$$57 = 2 \times 28 + 1$$

$$57 = 2 \times 2 \times 14 + 1$$

$$57 = 2 \times 2 \times 2 \times 7 + 1$$

$$57 = 2 \times 2 \times 2 \times (2 \times (2 \times 1 + 1) + 1) + 1$$

$$57 = 2 \times 2 \times 2 \times (2 \times 2 + 2 + 1) + 1$$

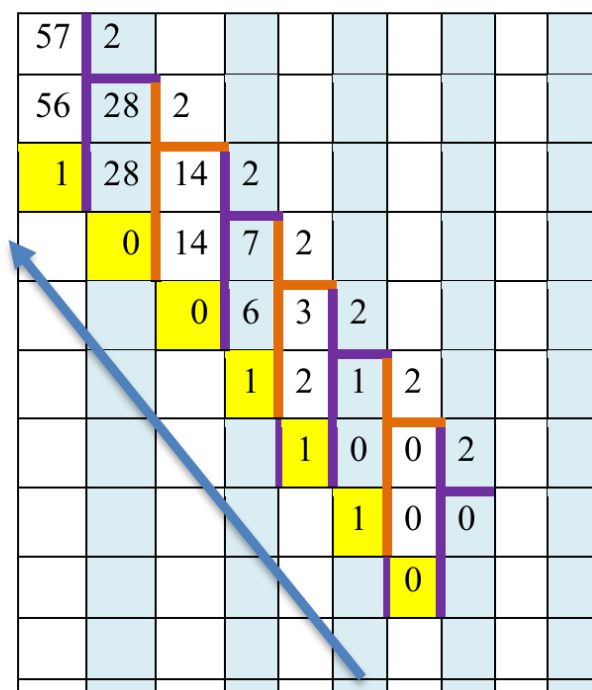
$$57 = 2^3 \times 2^2 + 2^3 \times 2^1 + 2^3 \times 2^0 + 1$$

$$57 = 2^5 + 2^4 + 2^3 + 2^0$$

Séquence	0	0	1	1	1	0	0	1
Position	7	6	5	4	3	2	1	0
Poids	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

$$57_{10} = 111001_2$$

Ou bien :



e. 100000_2 en base 10

Séquence	1	0	0	0	0	0
Position	5	4	3	2	1	0
Poids	2^5	2^4	2^3	2^2	2^1	2^0

$$100000_2 = (2^5)_{10}$$

$$100000_2 = (32)_{10}$$

Exercice 3 : décomposition

Décomposer les nombres binaires suivants en somme de puissances de 2, puis donner leur valeur en écriture décimale.

a. 10101_2

Séquence	1	0	1	0	1
Position	4	3	2	1	0
Poids	2^4	2^3	2^2	2^1	2^0

$$10101_2 = (1 \times 2^4 + 1 \times 2^2 + 1 \times 2^0)_{10}$$

$$10101_2 = (16 + 4 + 1)_{10}$$

$$10101_2 = (21)_{10}$$

b. 110010_2

Séquence	1	1	0	0	1	0
Position	5	4	3	2	1	0
Poids	2^5	2^4	2^3	2^2	2^1	2^0

$$110010_2 = (1 \times 2^5 + 1 \times 2^4 + 1 \times 2^1)_{10}$$

$$110010_2 = (32 + 16 + 2)_{10}$$

$$110010_2 = (50)_{10}$$

c. 10000001_2

$$10000001_2 = (1 \times 2^7 + 1 \times 2^0)_{10}$$

$$10000001_2 = (128 + 1)_{10}$$

$$10000001_2 = (129)_{10}$$

Exercice 4 : conversion décimal → binaire

a. 18 :

$$18 = 2 \times 9$$

$$18 = 2 \times (2 \times 4 + 1)$$

$$18 = 2 \times (2 \times 2 \times 2 + 1)$$

$$18 = 2 \times 2^3 + 2 \times 1$$

$$18 = 2^4 + 2^1$$

Séquence	1	0	0	1	0
Position	4	3	2	1	0
Poids	2^4	2^3	2^2	2^1	2^0

$$18_{10} = (10010)_2$$

Ou bien :

18	2					
18	9	2				
0	8	4	2			
	1	4	2	2		
		0	2	1	2	
			0	0	0	
				1		

b. 37 :

37	2					
36	18	2				
1	18	9	2			
	0	8	4	2		
		1	4	2	2	
			0	2	1	2
				0	0	0
					1	

$$37_{10} = (100101)_2$$

c. 64 :

64	2						
64	32	2					
0	32	16	2				
	0	16	8	2			
		0	8	4	2		
			0	4	2	2	
				0	2	1	2
					0	0	0
						1	

$$64_{10} = (1000000)_2$$

d. 91 :

91	2						
90	45	2					
1	44	22	2				
	1	22	11	2			
		0	10	5	2		
			1	4	2	2	
				1	2	1	2
					0	0	0
						1	

$$91_{10} = (1011011)_2$$

e. 127 :

127	2						
126	63	2					
1	62	31	2				
	1	30	15	2			
		1	14	7	2		
			1	6	3	2	
				1	2	1	2
					1	0	0
						1	

$$127_{10} = (1111111)_2$$

Exercice 5 : conversion binaire → décimal

a. 1110 :

$$1110_2 = (1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1)_2$$

$$1110_2 = (8 + 4 + 2)_{10}$$

$$1110_2 = (14)_{10}$$

b. 10010 :

$$10010_2 = (1 \times 2^4 + 1 \times 2^1)_{10}$$

$$10010_2 = (16 + 2)_{10}$$

$$10010_2 = (18)_{10}$$

c. 100000000 :

$$100000000_2 = (2^8)_{10}$$

$$100000000_2 = (256)_{10}$$

Exercice 6

Compléter le tableau en convertissant chaque nombre en sa représentation hexadécimale.

Décimal	10	11	12	13	14	15	16	31	255
Hexadécimal	A	B	C	D	E	F	10	1F	FF

255	16	
240	15	16
15	0	0
	15	



Exercice 7

Compléter la correspondance entre les chiffres hexadécimaux A à F et leur écriture binaire (sur 4 bits), puis convertir en binaire les nombres hexadécimaux 3A, 2F, FF et A5.

Hexadécimal	A	B	C	D	E	F
Binaire	1010	1011	1100	1101	1110	1111

Hexadécimal	3A	2F	FF	A5
Binaire	00111010	00101111	11111111	10100101

Exercice 8 : conversion binaire → hexadécimal

Binaire	10101111	11110000	10011101	11000011
Hexadécimal	AF	F0	9D	C3

Exercice 9 : Vrai/Faux

- 1°) Le système décimal possède dix chiffres. Vrai
- 2°) 102_2 existe. Faux
- 3°) 1111_2 vaut 15_{10} Vrai
- 4°) A_{16} vaut 11_{10} Faux
- 5°) En base 8, le chiffre 9 est autorisé. Faux
- 6°) Un chiffre hexadécimal correspond à 4 bits. Vrai

Exercice 10 : capacité

Déterminez le nombre de valeurs différentes que l'on peut représenter avec un codage sur 3, 4, 5, 8 et 16 bits. Justifiez votre réponse à l'aide d'une formule générale.

Si un codage comporte **n bits**, chaque bit ayant 2 possibilités, le nombre total de combinaisons est obtenu en multipliant les possibilités de chaque bit :

Sur 3 bits : $2^3 = 8$

On peut représenter 8 valeurs différentes de 0 à 111.

Sur 4 bits : $2^4 = 16$

On peut représenter 16 valeurs différentes.

Sur 5 bits : $2^5 = 32$

On peut représenter 32 valeurs différentes.

Sur 16 bits : $2^{16} = 65\,536$

On peut représenter 65 536 valeurs différentes.

Exercice 11 : valeur maximale

Déterminez la valeur maximale que l'on peut représenter avec un codage non signé sur 4, 5, 8 et 10 bits. Justifiez votre réponse à l'aide d'une formule générale.

Remarque : On suppose ici qu'il s'agit **d'un codage non signé** (c'est-à-dire que seules des valeurs positives ou nulles sont représentées).

Pour trouver la valeur maximale représentable avec un certain nombre de bits, il faut placer 1 dans tous les bits.

Par exemple, avec 4 bits, la plus grande valeur binaire est :

$$1111_2$$

Cette valeur correspond à :

$$8 + 4 + 2 + 1 = 15$$

On remarque que le nombre total de valeurs représentables sur n bits est :

$$2^n$$

Ces valeurs vont de 0 jusqu'à la valeur maximale. Comme on commence à compter à 0, la valeur maximale est donc :

$$2^n - 1$$

Avec 5 bits :

$$2^5 - 1 = 32 - 1 = 31$$

Avec 8 bits :

$$2^8 - 1 = 256 - 1 = 255$$

Avec 10 bits :

$$2^{10} - 1 = 1024 - 1 = 1023$$

Exercice 12 : Codage

On considère un codage binaire non signé sur 8 bits.

1°) Déterminez la valeur minimale et la valeur maximale représentables.

2°) Peut-on représenter le nombre 300 avec un codage sur 8 bits ?
Justifiez votre réponse.

3°) Si ce n'est pas possible, déterminez le nombre minimal de bits nécessaires pour représenter le nombre 300.

Avec un codage **non signé** sur **8 bits**, on peut représenter **256 valeurs différentes**, allant de **0** à **255**. La valeur minimale est donc **0** et la valeur maximale est **255**.

Le nombre **300** est supérieur à **255** ; il ne peut donc pas être représenté sur **8 bits**.

Pour représenter 300, on cherche le plus petit nombre de bits vérifiant :

$$2^n - 1 \geq 300$$

Comme **8 bits** permettent d'aller seulement jusqu'à **255** et que **9 bits** permettent d'aller jusqu'à **511**, il faut **9 bits** au minimum.

Exercice 13 :

Une civilisation extraterrestre possède **8 doigts** (au lieu de 10). Elle a donc choisi de compter en utilisant une base adaptée à son nombre de doigts.

1°) Quelle est la **base de numération** utilisée par cette civilisation ?

2°) Quels sont les **chiffres** utilisés dans cette base ?

3°) a. Écrivez les nombres de **0 à 20** dans cette base.

b. Convertissez les nombres décimaux **27**, **32** et **64** en base 8.



1°) Notre système de numération est en **base 10** car nous avons dix doigts.

Cette civilisation possède **8 doigts**. Il est donc logique qu'elle utilise la **base 8**, appelée **base octale**.

2°) Les chiffres utilisés sont 0, 1, 2, 3, 4, 5, 6, 7.

3°)

Décimal	Base 8
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	10
9	11
10	12
11	13
12	14
13	15
14	16
15	17
16	20
17	21
18	22
19	23
20	24
...	...
27	33
32	40
64	100

Exercice 14 : défi

Répondez aux questions suivantes en justifiant vos calculs lorsque cela est nécessaire.

1°) Déterminez la valeur en base 10 du nombre binaire 11111111_2 .

2°) Déterminez la valeur en base 10 du nombre hexadécimal $FFFF_{16}$.

3°) Combien de valeurs différentes peut-on représenter avec 32 bits ?

$$1^\circ) 11111111_2 = (1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0)_{10}$$

$$11111111_2 = (128 + 64 + 32 + 16 + 8 + 4 + 2 + 1)_{10}$$

$$11111111_2 = (255)_{10}$$

$$2^\circ) FFFF_{16} = (15 \times 16^3 + 15 \times 16^2 + 15 \times 16^1 + 15 \times 16^0)_{10}$$

$$FFFF_{16} = (61440 + 3840 + 240 + 15)_{10}$$

$$FFFF_{16} = 65\,535_{10}$$

$$3^\circ)$$

$$2^{32} = 4\,294\,967\,296$$

Exercice 15 : Synthèse – Représentation des nombres

Une machine utilise des mots de 16 bits. Répondez aux questions suivantes en justifiant vos réponses lorsque cela est nécessaire.

1°) Combien de valeurs différentes peut-on représenter avec 16 bits ?

2°) Quel est le plus grand entier non signé que l'on peut représenter sur 16 bits ?

3°) Convertissez le nombre binaire **110101101011110**₂ en hexadécimal.

4°) Convertissez ensuite ce nombre en décimal.

5°) Expliquez pourquoi les ordinateurs utilisent le système binaire.

$$1^\circ) 2^{16} = 65\,536$$

On peut représenter 65 536 valeurs avec 16 bits.

2°)

Le plus grand nombre est obtenu lorsque les 16 bits valent 1 :

$$1111111111111111_2$$

Sa valeur est $2^{16} - 1 = 65\,535$

Ces valeurs commencent à **0** et finissent à la plus grande valeur possible :

Valeur décimale	Valeur binaire (16 bits)
0	0000000000000000
1	0000000000000001
2	0000000000000010
3	0000000000000011
4	0000000000000100
.....	
65 535	1111111111111111

3°)

$$1101011010111110_2 = (D6BE)_{16}$$

4°)

$$(D6BE)_{16} = (13 \times 16^3 + 6 \times 16^2 + 11 \times 16^1 + 14 \times 16^0)_{10}$$

$$(D6BE)_{16} = (53\,248 + 1\,536 + 176 + 14)_{10}$$

$$(D6BE)_{16} = (54\,974)_{10}$$

5°)

Les ordinateurs sont constitués de **milliards de transistors**. Un transistor est un composant électronique qui ne possède que **deux états stables** :

- **État 0** : le courant ne passe pas (ou tension faible).
- **État 1** : le courant passe (ou tension élevée).

Comme les transistors ne peuvent prendre que **deux états**, il est naturel de représenter les informations avec uniquement les chiffres **0** et **1**. C'est le **système binaire**.

Toutes les informations (nombres, textes, images, sons, vidéos, etc.) sont ensuite codées sous forme de suites de 0 et de 1, ce qui permet à l'ordinateur de les stocker, de les transmettre et de les traiter facilement.