

Représentation des nombres

Arnaud Bannier
Nicolas Bodin
Aurélien Texier

1. La théorie

Exercice 1. Entiers naturels

Complétez le tableau suivant donnant la représentation d'un même nombre dans différentes bases. La première ligne vous est donnée en tant qu'exemple.

Binaire	Octal	Décimal	Hexadécimal
10110	26	22	16
101011			
	57		
		61	
			3C

Exercice 2. Entiers relatifs

Complétez le tableau ci-dessous donnant la valeur d'un même nombre dans sa représentation binaire classique pour la valeur absolue du nombre, et dans ses représentations complément à 1 et à 2. Les nombres binaires sont supposés être représentés sur un octet.

Binaire	Complément à 1	Complément à 2	Décimal
00101101	11010010	11010011	-45
01100101			
	10110111		
		11100100	
			-86

Exercice 3. Opérations

Posez les opérations suivantes en utilisant une notation binaire. Les nombres négatifs sont exprimés en complément à 2.

3.1) $43 + 38$

3.2) $43 - 38$

3.3) $38 - 43$

3.4) 43×38

Exercice 4. Rationnels

Complétez le tableau ci-dessous donnant la représentation d'un même nombre réel dans sa représentation décimale et dans sa représentation donnée par la norme IEEE754. Rappelons que cette dernière décompose un rationnel x dans un triplet de valeurs binaires (s, e, m) résolvant $x = (s.m.10^e)_2$. avec :

- s donnant le signe sur 1 bit,
- e donnant l'exposant sur 8 bits ,
- m donnant la mantisse sur 23 bits.

Les valeurs sont regroupés par paquets de 4 bits afin d'améliorer la lisibilité.

Rationnel	Signe	Exposant	Mantisse
17.5	0	1000 0011	0001 1000 0000 0000 0000 000
		1000 0011	1001 1000 1010 0000 0000 000
		1000 0100	0101 0011 1000 1100 0000 000
14.171875			
0.1			

Exercice 5. Logique binaire¹

Soit un programme contenant les instructions suivantes :

```
1  int i = 8;
2  int j = 5;
3  float x = 0.005;
4  float y = -0.01;
5  char c = 'c';
6  char d = 'd';
```

Déterminez la valeur de chacune des expressions suivantes. Vérifiez le résultat à l'aide de votre ordinateur.

- 5.1) $(3*i - 2*j) \% (2*d - c)$
- 5.2) $2*((i/5) + (4*(j-3)) \% (i + j - 2))$
- 5.3) $i \leq j$
- 5.4) $j \neq 6$
- 5.5) $c == 99$
- 5.6) $5*(i + j) > 'c'$
- 5.7) $(i > 0) \&\& (j < 5)$
- 5.8) $((i > 0) \|\ (j < 5)) \&\& (x > 0)$
- 5.9) $i \& j$
- 5.10) $i \mid j$

2. ...et la pratique

Dans cette section nous allons manipuler des entiers au niveau binaire afin de comprendre leur fonctionnement dans la machine. Nous supposons que tous les entiers manipulés ici sont des entiers signés au format `int`.

1. Exercice issu de <http://www.montefiore.ulg.ac.be>

Exercice 6. Exercices d'introduction

Dans l'exercice suivant, utilisez au maximum les manipulations binaires. Le symbole % et la fonction `pow()` sont interdits. Soient `n` et `p` deux entiers. Rédigez un code permettant de ...

- 6.1) renvoyer 0 si l'entier `n` est pair, 1 sinon.
- 6.2) multiplier l'entier `n` par 2^p .
- 6.3) donner le reste de la vision de `n` par 2^p .
- 6.4) renvoyer le nombre de bits à 1 de `n`.
- 6.5) afficher `n` en base 8.

Exercice 7. Conversion en négatif

Vous devez réaliser une fonction prenant en paramètres un entier et retournant l'opposé de cet entier. Par exemple, en entrant l'entier 6, votre fonction retourne -6, et inversement. Pour cela, il vous est interdit de multiplier l'entier par la valeur (-1), de soustraire 2 fois sa valeur etc. L'objectif est ici de réaliser des manipulations binaires. Relisez donc votre cours afin de trouver l'algorithme répondant à ce problème.

Exercice 8. Manipulation des flottants

La norme IEEE754 permet de définir un flottant à l'aide d'une union. L'objectif de cette construction est d'atteindre rapidement l'ensemble des bits de la mantisse ou de l'exposant.

```
1  typedef union ieee754_float
2  {
3      float f;
4      struct {
5          unsigned int mantisa : 23;
6          unsigned int exponent : 8;
7          unsigned int sign : 1;
8      } ieee;
9  } float_cast;
```

Comme vous l'avez vu en cours, grâce à cette écriture des nombres, il est possible manipuler autant de rationnels différents entre -1 et 1 que dans le reste de \mathbb{Q} . Cela signifie que « l'espace » entre deux nombres qu'il est possible de stocker de façon exacte n'est pas le même suivant la valeur du rationnel. Le but de cet exercice est de mesurer cet espace en fonction de la valeur du nombre, ce qui nous indiquera la précision qu'il est possible d'avoir avec des `float`.

- 8.1) Dans un premier temps, codez la fonction `Printb()` prenant en paramètres un entier à afficher au format binaire. Vous passerez également en second argument le nombre de bits à afficher.
- 8.2) Utilisez votre fonction précédente afin de coder la fonction `Print_ieee()` permettant d'afficher l'ensemble des champs de la structure présentée ci-dessus.
- 8.3) Testez votre programme sur les exemples du cours afin de vérifier que votre code est correct. Vous passerez le rationnel à afficher en argument de la ligne de commande. La fonction `atof()` vous sera d'une grande aide.
- 8.4) Codez la fonction `Precision()` prenant en paramètres un flottant et retournant la différence de valeur entre ce flottant et son suivant immédiat qu'il est possible de stocker avec la norme IEEE. Vous afficherez l'ensemble des valeurs en question en binaire et en décimal.
- 8.5) Quelle est la meilleure précision atteignable ? La pire ?