Numérique et sciences informatiques
Première

B. Représentation de l'information
6. Représentation des nombres « à virgule » :
les nombres flottants

I Représentation en mémoire des nombres flottants

Commençons par une première remarque importante. Si on dispose d'un espace mémoire fini, il est impossible de créer un système de représentation qui permette de représenter *tous* les nombres réels compris dans un intervalle donné, même si cet intervalle est borné.

I.1 Écriture scientifique décimale et binaire

a) Rappel sur l''écriture scientifique décimale

• Pour tout nombre décimal *non nul* D, il existe une unique façon d'écrire D sous la forme suivante :

$$sX.10^N$$

où :

* s est le **signe** : $+$ ou $-$;
* X est **la mantisse** : un nombre décimal appartenant à l'intervalle $[1\ ;\ 10[$;
* N est **l'exposant de 10** : un entier *relatif*.

Exemples :

$$486{,}986 = +4{,}86986.10^2$$
$$-0{,}004058 = -4{,}058.10^{-3}$$

• Il faut remarquer que le nombre 0 ne peut pas être écrit sous la forme d'une écriture scientifique puisque :

* $X \neq 0$ puisque $X \in [1\ ;\ 10[$
* $10^N \neq 0$pour tout entier relatif N.

b) L'écriture scientifique binaire

En écriture scientifique binaire, on représente certains nombres relatifs (ceux que l'on peut écrire sous la forme d'une écriture à virgule binaire) sous la forme scientifique binaire:

$$sM.2^E$$

où :

* s est **le signe** : $+$ ou $-$;
* M est **la mantisse** : un nombre à virgule binaire appartenant à l'intervalle $[1\ ;\ 2[$;
* E est **l'exposant de 2** : un entier relatif.

• Exercice : quel est le nombre dont la représentation en écriture scientifique binaire est la suivante ?

$$-\underline{1{,}101}_2.2^5$$

I.2 Simple précision et double précision

Pour représenter un nombre non entier dans la mémoire d'un ordinateur, on utilise l'écriture scientifique binaire. La norme internationale IEEE 754 élaborée en 1984 fixe les règles de représentation des nombres à virgule flottante. Cette norme définit l'expression du

signe, de l'exposant et de la mantisse selon l'espace mémoire dont on dispose. On distingue principalement la représentation en **simple précision** sur 4 octets et en **double précision** sur 8 octets.

a) En « simple précision »

On dispose de 32 bits (4 octets).

* le **premier bit** (de poids fort) exprime le signe : 0 pour le signe $+$ et 1 pour le signe $-$;
* les **8 bits suivants** expriment la valeur de l'exposant E ;
* les **23 derniers bits** expriment la matisse M.

b) En « double précision »

On dispose de 64 bits (8 octets).

* le **premier bit** pour le signe ;
* les **11 bits suivants** pour l'exposant E ;
* les $\mathbf{52}$ **derniers bits** pour la matisse M.

C'est cette représentation qui est notamment utilisée pour les nombres dits « flottants » (de type **`float)`** en Python.

I.3 Représentation de l'exposant E

• Dans la norme IEEE 754, on ne représente pas l'exposant E, qui est un entier positif ou négatif, par la méthode du « complément à deux ». On l'exprime par un entier positif N à partir duquel on déduit l'exposant E, négatif ou positif E grâce à la convention suivante :

$$E = N - d$$

où d est un nombre choisi pour permettre de représenter autant de nombres positifs que de nombres négatifs.

• En simple précision, on dispose de 8 bits donc N est compris entre 0 et 255. On choisit alors $d = 127$, soit $2^7 - 1$.

• Ceci permettrait en théorie d'obtenir l'exposant E compris entre -127 et 128. En pratique, on réserve les cas où N est égal à 0 et à 255 pour exprimer certains cas particuliers comme $+0$, -0, $+\infty$, $-\infty$ et NaN. E est donc *compris entre* -126 *et* 127.

• On peut alors représenter les nombres décimaux compris approximativement entre 10^{-38} et 10^{38}.

• En double précision, on dispose de 11 bits, on choisit donc $d = 1023$ soit $2^{10} - 1$. On peut alors représenter les nombres décimaux compris approximativement entre 10^{-308} et 10^{308}.

I.4 Représentation de la mantisse M

• Puisque la mantisse est toujours un nombre compris entre 1 inclus et 2 exclu, on utilise tous les bits pour représenter *les chiffres après la virgule* (bien sûr en écriture binaire). Ainsi, si les chiffres de la mantisse (en simple précision) sont :

$$b_1 b_2 \dots b_{23}$$

Alors la valeur exprimée par la mantisse est :

$$M = 1 + b_1 \times 2^{-1} + b_2 \times 2^{-2} + \cdots + b_{23} \times 2^{-23}$$

I.5 Les valeurs particulières

Le tableau ci-dessous indique les conventions utilisées pour les quelques valeurs particulières :

Valeur représentée	Signe	N	Mantisse
Zéro (+)	0	0	0
Zéro (-)	1	0	0
$+\infty$	0	255	0
$-\infty$	1	255	0
NaN : *Not a Number*	0	255	non nulle