

Corrigé – Représentation des nombres réels

Exercice 1

$$5,84375_{10} = 101,11011_2 \text{ et } 0,1_{10} = 0,0001\,1001\,1001\,1001\,1001\ldots_2 = 0,0\overline{0011}.$$

Exercice 2

$$123\,000\,000 = 1,23 \times 10^8 \text{ et } -0,000\,321 = -3,21 \times 10^{-4}$$

Exercice 3

1. En écriture scientifique en base 2 :

$$14,5 = m \times 2^e \Rightarrow m = 14,25/2^e$$

Or :

$$14,25/2 = 7,125$$

$$7,125/2 = 3,5625$$

$$3,5625/2 = 1,78125$$

$$\text{On a trois divisions par deux donc } e = 3 \text{ et donc } m = 14,25/2^3 \Rightarrow 14,25 = 1,78125 \times 2^3$$

2. $e + 127 = 3 + 127 = 130 = 1000\,0010_2$

et

$$m = 1,78125$$

On garde le 1 de côté et on convertit la partie décimale en binaire :

$$0,78125 \times 2 = \boxed{1} + 0,5625$$

$$0,5625 \times 2 = \boxed{1} + 0,125$$

$$0,125 \times 2 = \boxed{0} + 0,25$$

$$0,25 \times 2 = \boxed{0} + 0,5$$

$$0,5 \times 2 = \boxed{1} + 0$$

$$\Rightarrow m = 1,11001$$

Exercice 4

$$128 = m \times 2^e$$

$$m = 128/2^e \text{ or } 128 = 1 \times 2^7$$

Le nombre est positif donc sa représentation commence par 0. On calcule $e+127 = 7+127 = 134 = 1000\,0110_2$ et il n'y a pas de chiffres après la virgule donc toutes les valeurs de la mantisse sont à 0.

On obtient donc que 128 est représenté sur 32 bits par $0\,1000\,0110\,000\,0000\,0000\,0000\,0000\,0000$

Exercice 5

$$0\,1000\,0010\,110\,0100\,0000\,0000\,0000\,0000$$

Exercice 6

a) 1 0111 1110 111100000000000000000000

- commence par 1 donc le signe est négatif
- $e + 127 = 0111\ 1110_2 \Rightarrow e = 0111\ 1110_2 - 127 = 126 - 127 = -1$
- $m = 111100000000000000000000 = 2^{-1} + 2^{-2} + 2^{-3} + 2^{-4} = 0,9375$

En suivant la norme, on peut donc déduire que la valeur cherchée est $-1,9375 \times 2^{-1} = \boxed{-0,96875}$

b) 0 1000 0011 111000000000000000000000

- commence par 0 donc le signe est positif
- $e = 1000\ 0011_2 - 127 = 131 - 127 = 4$
- $m = 111000000000000000000000 = 2^{-1} + 2^{-2} + 2^{-3} = 0,875$

En suivant la norme, on peut donc déduire que la valeur cherchée est $+1,875 \times 2^4 = \boxed{+30}$

Exercice 7

1. Le script ne se termine pas parce que la variable x ne vaut jamais exactement 0 étant donné que la valeur 0,1 est représentée de manière approximative dans l'ordinateur.
2. Pour pallier ce problème, on peut remplacer != par > comme suit :

```
1 x = 10
2
3 while x > 0 :
4     x = x - 0.1
5     print(x)
```

Exercice 8

`print(2**1024)` affiche un entier très grand sans problème tandis que `print(2.0**1024)` génère une erreur : `OverflowError: (34, 'Numerical result out of range')`

Le résultat est trop grand pour l'espace mémoire dédié au stockage des flottants (64 bits en Python).