

Python : Cours n° 4 : les nombres

Nombres décimaux :

Les décimaux sont appelés des flottants (float) :

$$2 \times 5 = 10$$

$$8 / 2 = 4.0$$

$$8 // 2 = 4 \text{ et } 8 // 3 = 2 \quad \rightarrow \text{division entière (quotient de la division euclidienne)}$$

$$9 \% 2 = 1 \quad \rightarrow \text{reste de la division euclidienne}$$

Puissances : → des calculs simples

$$4 \times 3 = 4^3 = 64$$

→ pas de taille max pour les entiers, cela dépend de la mémoire physique de l'ordinateur

↑ : rappeler la dernière instruction

$$2.0^{1023} : \quad 8.98846567431158e+307$$

$$2.0^{1024} : \quad \text{saturation overflow error} \rightarrow 2^{1024} \text{ existe}$$

→ taille max : environ 10^{307}

En base 2 : stockage de 52 chiffres possible après la virgule

Tests et booléens :

Test :	$3 * 2 == 6$	réponse TRUE
	$2 * 0.1 == 0.2$	réponse TRUE
	$3 * 0.1 == 0.3$	réponse FALSE

Explications !!!

En base 10 :

$$\begin{array}{r} 1 \quad | \quad 3 \\ 10 \quad | \quad 0,33... \\ \hline 10 \quad | \end{array}$$

En base 16 :

$$\begin{array}{r} 1 \quad | \quad 5 \\ (10)_{16} \quad | \quad 0,33... \\ \hline (10)_{16} \quad | \end{array}$$

Ainsi $\frac{1}{3} = (0,333...)_{10}$ et $\frac{1}{5} = (0,333...)_{16}$

Or $3 = (0011)_2$ donc $\frac{1}{5} = (0,001100110011...)_{2}$

Or seules 52 décimales sont stockées en mémoire, on perd les dernières décimales, ce qui induira une erreur.

→ dans les tests, demander si la différence est inférieure à 10^{-15}

L'arithmétique sur les flottants est très compliquée

$1 + 2** -52 == 1$	réponse FALSE	→ 1, 0...1 <small>52bits</small>
--------------------	---------------	-------------------------------------

$1 + 2** -53 == 1$	réponse TRUE	→ 1, 0...01 <small>52bits</small>
--------------------	--------------	--------------------------------------

Bibliothèques :

`Sqrt(25)` → ERROR

→ il faut appeler d'abord la bibliothèque math :

From math import * (en début de programme, s'il n'y a qu'une seule bibliothèque)

`math.sqrt(25)` → 5.0

`math.cos(math.pi)` → -1.0

NB : `help(math.sqrt)` → infos sur l'option math

NB : `dir(math)` → liste des objets de math

From math import floor, ceil → arrondis inférieur et supérieur

Floor(43.5) → 43

Ceil(43.5) → 44