Attention aux opérations effectuées par les ordinateurs sur les nombres décimaux

Lorsqu'on effectue des calculs simples dans une console Python, on peut avoir des surprises comme le montre les deux copies d'écrans suivantes.

```
>>> 10.625+10.625
21.25
>>> 21.25+10.625
31.875
>>> 10.625+10.625+10.625
31.875
```

Figure 1

```
>>> 0.1+0.1
0.2
>>> 0.2+0.1
0.30000000000000004
>>> 0.1+0.1+0.1
0.30000000000000004
```

Figure 2

Dans la série de calculs de gauche, tout se passe bien.
Par contre, à droite, le calcul de 0,1+0,2 ne donne pas un résultat satisfaisant.

Que se passe-t-il dans la deuxième série de calculs ?

Dans la suite, on va tenter de donner des éléments de réponse.

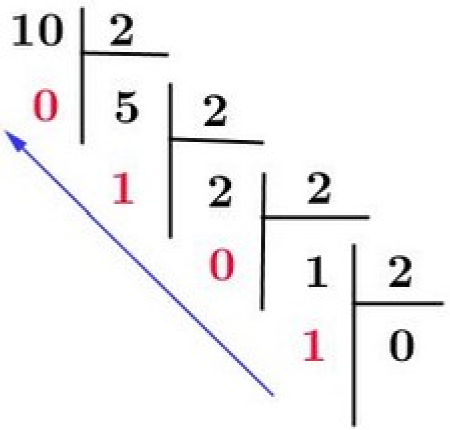
Dans un premier temps, rappelons que le résultat suivant.

> Si x est un nombre décimal, alors x peut s'écrire sous la forme $x=\sum_{n\in\mathbb{Z}} a_n\times 2^n$ où $a_n\in\{0;1\}$

C'est la série des nombres a_n qui va permettre à l'ordinateur de coder un nombre décimale.

Exemple : Prenons par exemple x=10,625 …. que l'on devrait d'ailleurs écrire $(10{,}625)_{10}$.
On a 10,625=10+0,625 ou encore $(10{,}625)_{10}=(10)_{10}+(0{,}625)_{10}$

Décomposition en binaire de 10



$$\begin{aligned}
&10=2\times5+0\\
\Leftrightarrow\ &10=2\times(2\times2+1)+0\\
\Leftrightarrow\ &10=2\times(2\times(2\times1+0)+1)+0\\
\Leftrightarrow\ &10=2\times(2^2+1)+0\\
\Leftrightarrow\ &10=2^3+2
\end{aligned}$$

donc $10=1\times2^3+0\times2^2+1\times2^1+0\times2^0$
donc $(10)_{10}=(1010)_2$

Décomposition en binaire de 0,625

$$0{,}625\times2=1{,}25$$
$$0{,}25\times2=0{,}5$$
$$0{,}5\times2=1$$

$$\begin{aligned}
&0{,}625=\frac{1{,}25}{2}\\
\Leftrightarrow\ &0{,}625=\frac{1}{2}+\frac{0{,}25}{2}\\
\Leftrightarrow\ &0{,}625=\frac{1}{2}+\frac{0{,}25\times2}{2\times2}\\
\Leftrightarrow\ &0{,}625=\frac{1}{2}+\frac{0{,}5}{4}\\
\Leftrightarrow\ &0{,}625=\frac{1}{2}+\frac{0{,}5\times2}{4\times2}\\
\Leftrightarrow\ &0{,}625=\frac{1}{2}+\frac{1}{2^3}
\end{aligned}$$

donc $0{,}625=\mathbf{1}\times\frac{1}{2}+\mathbf{0}^{*}\times\frac{1}{2^2}+\mathbf{1}\times\frac{1}{2^3}$
donc $(0{,}625)_{10}=(101)_2$

Finalement, on a $(10{,}625)_{10}=(1010{,}101)_2$
Ainsi, si on simplifie un peu, le nombre 10,625 est « codé » par un ordinateur par 1010,101.

En fait c'est un peu plus compliqué que cela mais cette représentation suffit à expliquer le résultat donné par Python dans la figure 2.

Que se passe-t-il dans le cas de 0,1 ou plutôt de $(0{,}1)_{10}$?

$0{,}1\times2=0{,}2$
$0{,}2\times2=0{,}4$
$0{,}4\times2=0{,}8$
$0{,}8\times2=1{,}6$
$0{,}6\times2=1{,}2$ (on ne garde que la partie décimale de 1,6 : voir ci-dessous)
$0{,}2\times2=0{,}4$
$0{,}4\times2=0{,}8$
$0{,}8\times2=1{,}6$ ....etc….

$$0{,}1=\frac{0{,}1\times2}{2}$$
$$\Leftrightarrow 0{,}1=\frac{0{,}2}{2}$$
$$\Leftrightarrow 0{,}1=\frac{0{,}2\times2}{2^2}$$
$$\Leftrightarrow 0{,}1=\frac{0{,}4\times2}{2^3}$$
$$\Leftrightarrow 0{,}1=\frac{0{,}8\times2}{2^4}$$
$$\Leftrightarrow 0{,}1=\frac{1{,}6}{2^4}$$
$$\Leftrightarrow 0{,}1=\frac{1+0{,}6}{2^4}$$
$$\Leftrightarrow 0{,}1=\frac{1}{2^4}+\frac{0{,}6}{2^4}$$
$$\Leftrightarrow 0{,}1=\frac{1}{2^4}+\frac{0{,}6\times2}{2^5}$$
$$\Leftrightarrow 0{,}1=\frac{1}{2^4}+\frac{1{,}2}{2^5}$$
$$\Leftrightarrow 0{,}1=\frac{1}{2^4}+\frac{1+0{,}2}{2^5}$$
$$\Leftrightarrow 0{,}1=\frac{1}{2^4}+\frac{1}{2^5}+\frac{0{,}2}{2^5}$$
$$\Leftrightarrow 0{,}1=\frac{1}{2^4}+\frac{1}{2^5}+\frac{0{,}2\times2^3}{2^5\times2^3}$$
$$\Leftrightarrow 0{,}1=\frac{1}{2^4}+\frac{1}{2^5}+\frac{1{,}6}{2^8}$$
$$\Leftrightarrow 0{,}1=\frac{1}{2^4}+\frac{1}{2^5}+\frac{1}{2^8}+\frac{0{,}6}{2^8}$$
$$\Leftrightarrow 0{,}1=\frac{1}{2^4}+\frac{1}{2^5}+\frac{1}{2^8}+\frac{1{,}2}{2^9}$$
$$\Leftrightarrow 0{,}1=\frac{1}{2^4}+\frac{1}{2^5}+\frac{1}{2^8}+\frac{1}{2^9}+\frac{0{,}2}{2^9}$$

...etc ….

ainsi $(0{,}1)_{10}=(0{,}00011001100\,...)_2$ ou encore $(0{,}1)_{10}=(0{,}000\underline{1100})_2$

Et pour 0,2 ou plutôt de $(0{,}2)_{10}$?

$0{,}2\times2=0{,}4$
$0{,}4\times2=0{,}8$
$0{,}8\times2=1{,}6$
$0{,}6\times2=1{,}2$ (on ne garde que la partie décimale de 1,6 : voir ci-dessous)
$0{,}2\times2=0{,}4$
$0{,}4\times2=0{,}8$
$0{,}8\times2=1{,}6$ ….etc….

$$0{,}2=\frac{0{,}2\times2}{2}$$
$$\Leftrightarrow 0{,}2=\frac{0{,}4}{2}$$
$$\Leftrightarrow 0{,}2=\frac{0{,}4\times2}{2^2}$$
$$\Leftrightarrow 0{,}2=\frac{0{,}8\times2}{2^3}$$
$$\Leftrightarrow 0{,}2=\frac{1{,}6}{2^3}$$
$$\Leftrightarrow 0{,}2=\frac{1+0{,}6}{2^3}$$
$$\Leftrightarrow 0{,}2=\frac{1}{2^3}+\frac{0{,}6}{2^3}$$
$$\Leftrightarrow 0{,}2=\frac{1}{2^3}+\frac{0{,}6\times2}{2^4}$$
$$\Leftrightarrow 0{,}2=\frac{1}{2^3}+\frac{1{,}2}{2^4}$$
$$\Leftrightarrow 0{,}2=\frac{1}{2^3}+\frac{1+0{,}2}{2^4}$$
$$\Leftrightarrow 0{,}2=\frac{1}{2^3}+\frac{1}{2^4}+\frac{0{,}2}{2^4}$$
$$\Leftrightarrow 0{,}2=\frac{1}{2^3}+\frac{1}{2^4}+\frac{0{,}2\times2^3}{2^4\times2^3}$$
$$\Leftrightarrow 0{,}2=\frac{1}{2^3}+\frac{1}{2^4}+\frac{1{,}6}{2^7}$$
$$\Leftrightarrow 0{,}2=\frac{1}{2^3}+\frac{1}{2^4}+\frac{1}{2^7}+\frac{0{,}6}{2^7}$$
$$\Leftrightarrow 0{,}2=\frac{1}{2^3}+\frac{1}{2^4}+\frac{1}{2^7}+\frac{1{,}2}{2^8}$$
$$\Leftrightarrow 0{,}2=\frac{1}{2^3}+\frac{1}{2^4}+\frac{1}{2^7}+\frac{1}{2^8}+\frac{0{,}2}{2^8}$$

...etc...

ainsi $(0{,}2)_{10}=(0{,}0011001100\,...)_2$ ou encore $(0{,}2)_{10}=(0{,}\underline{0011})_2$ **(2)**

Remarque :
Dans les écritures binaires de $(0,1)_{10}$ et $(0,2)_{10}$, les séquences respectivement 1100 et 0011 se répètent indéfiniment. Et c'est là le problème car l'ordinateur n'a pas une mémoire infinie. Il doit donc tronquer cette écriture binaire.

Prenons, par exemple, un ordinateur qui tronque l'écriture binaire d'un nombre à la 24 ième décimale, ce qui en réalité est bien supérieur. Lorsqu'on effectue l'opération $(0,1)_{10}+(0,1)_{10}$, cet ordinateur n'obtiendra pas la même chose qu'un élève de CM1.

Elève de CM1	**Un ordinateur qui tronque à la 24ième décimale**
Opération en écriture décimale : $(0,1)_{10}+(0,1)_{10}=(0,2)_{10}$ Opération en écriture binaire : $(0,000\underline{1100})_2+(0,000\underline{1100})_2=(0,\underline{0011})_2$ en effet, $\begin{array}{r} 0,0001100110011001100\ldots\ldots \\ +\ 0,0001100110011001100\ldots\ldots \\ \hline 0,0011001100110011001\ldots\ldots \end{array}$ or $(0,\underline{0011})_2=(0,2)_{10}$ d'après **(2)**	$\begin{array}{r} 0,00011001100110011001100 \\ +\ 0,00011001100110011001100 \\ \hline 0,00110011001100110011000 \end{array}$ or si on écrit le résultat R de cette opération sous forme décimale, on obtient : $R=\frac{1}{2^3}+\frac{1}{2^4}+\frac{1}{2^7}+\frac{1}{2^8}+\frac{1}{2^{11}}+\frac{1}{2^{12}}+\frac{1}{2^{15}}+\frac{1}{2^{16}}+\frac{1}{2^{19}}+\frac{1}{2^{20}}$ $\Leftrightarrow R=\frac{2^{17}+2^{16}+2^{13}+2^{12}+2^9+2^8+2^5+2^4+2+1}{2^{20}}$ $\Leftrightarrow R=\frac{209715}{2^{20}}$ donc $R\approx 0,199$

On constate donc que cette ordinateur ne peut pas afficher la réponse attendue par l'élève de CM1.

Il en est de même pour nos ordinateurs.

Ce n'est donc pas la faute de Python si l'on n'obtient pas ce que l'on veut lorsqu'on effectue des calculs sur les nombres décimaux.