Séquence :	Numérique et Sciences Informatiques	Thème :
Représentation binaire approximative d'un réels : les flottants.		Représentation des données (types de base)

I – Représentation de la partie à droite de la virgule d'un nombre à virgule

En notation décimale, les chiffres à gauche de la virgule représentent des entiers, des dizaines, des centaines, des milliers, … etc, et ceux à droite de la virgule représentent des dixièmes, centièmes, millièmes, etc.

Ainsi : $3{,}8125_{(10)} = 3\times10^{0} + 8\times10^{-1} + 1\times10^{-2} + 2\times10^{-3} + 5\times10^{-4}$

De la même façon, pour un nombre à virgule en base 2, on utilise les puissances négatives de 2.

Observe cet exemple :

$$\begin{aligned} 11{,}1101_{(2)} &= 1\times2^{1} + 1\times2^{0} + 1\times2^{-1} + 1\times2^{-2} + 0\times2^{-3} + 1\times2^{-4} \\ &= 2+1+\frac{1}{2}+\frac{1}{4}+0+\frac{1}{16} \\ &= 2+1+0{,}5+0{,}25+0+0{,}0625 \\ &= 3{,}8125_{(10)} \end{aligned}$$

De la même manière, convertis le nombre suivant en base décimale : $(101{,}010101)_{2,}$

II – Conversion en binaire de la partie à droite de la virgule

1. Observez le schéma ci-contre :

Que permet-il de faire ?

$$0{,}375\times10 = \boxed{3}\ \underbrace{,75}$$
$$0{,}75\ \times10 = \boxed{7}\ \underbrace{,5}$$
$$0{,}5\ \ \times10 = \boxed{5}\ ,0$$

2. Faites la même chose avec 0,42 et 0,8769. Comment sait-on que l'on a terminé ?

3. Pour obtenir en binaire la partie à droite de la virgule, on procède de la même façon, mais en multipliant par 2 à chaque étape car on travaille alors en base 2.

- ***a.*** En adaptant le procédé ci-dessus, déterminer la partie à droite de la virgule en base 2 de 0,375.
- ***b.*** Vérifiez que le résultat obtenu en base 2, lorsqu'il est converti en base 10 redonne bien 0,375.
- ***c.*** Faire la même chose avec 0,625 puis vérifier le résultat obtenu.
- ***d.*** Faire de même avec 0,1. Que remarquez-vous ?

$$0{,}375\times2 = \boxed{0}\ \underbrace{,75}$$
$$0{,}75\ \times2 = \boxed{1}\ \underbrace{,5}$$
$$0{,}5\ \ \times2 = \boxed{1}\ ,0$$

4. a. On souhaite réaliser un algorithme en langage naturel qui effectue cette conversion. En observant les conversions faites précédemment, écrire en langage naturel les instructions qui sont répétées.

b. Peut-on prévoir à l'avance combien de fois elles vont être répétées ? Si non, comment peut-on savoir à quel moment il faudra arrêter de les répéter ?

c. Rédiger un algorithme en langage naturel pour effectuer la conversion.

5. Voilà un programme Python qui met en œuvre cet algorithme.
Les bits sont stockés dans une chaîne de caractères.

```
from decimal import *

n = Decimal(input("Entrer la partie décimale : "))
en_binaire = ''

i = 0
while (n != 0) and (i < 64):
    n = n*2
    if n >= 1:
        en_binaire = en_binaire + '1'
        n = n - 1
    else:
        en_binaire = en_binaire + '0'
    i = i+1

print(en_binaire)
```

a. Testez le programme avec les valeurs du début de l'activité.
b. Modifiez le programme pour qu'il affiche les calculs intermédiaires comme ci-dessous :

$$0{,}375\times 2 = 0\ ,75$$
$$0{,}75\ \times 2 = 1\ ,5\ .$$
$$0{,}5\ \ \times 2 = 1\ ,0$$

6. Reprenez le programme Python précédent et remplacez **Decimal** par **float.**
Reprenez les tests précédents. Que remarquez-vous ?

---

5. Voilà un programme Python qui met en œuvre cet algorithme.
Les bits sont stockés dans une chaîne de caractères.

```
from decimal import *

n = Decimal(input("Entrer la partie décimale : "))
en_binaire = ''

i = 0
while (n != 0) and (i < 64):
    n = n*2
    if n >= 1:
        en_binaire = en_binaire + '1'
        n = n - 1
    else:
        en_binaire = en_binaire + '0'
    i = i+1

print(en_binaire)
```

a. Testez le programme avec les valeurs du début de l'activité.
b. Modifiez le programme pour qu'il affiche les calculs intermédiaires comme ci-dessous :

$$0{,}375\times 2 = 0\ ,75$$
$$0{,}75\ \times 2 = 1\ ,5 \qquad \text{donc } 0{,}375 = 0{,}011.$$
$$0{,}5\ \ \times 2 = 1\ ,0$$

6. Reprenez le programme Python précédent et remplacez **Decimal** par **float.**
Reprenez les tests précédents. Que remarquez-vous ?