REPRÉSENTATION DES NOMBRES RÉELS - CORRECTION

Exercice 1 : codage en virgule fixe

a) Pour la partie entière : $(4)_{10}=(100)_2$
Pour la partie décimale : $(0,125)_{10}$:

$$0,125\times 2=0,25=\mathbf{0}+0,25$$
$$0,25\times 2=0,5=\mathbf{0}+0,5$$
$$0,5\times 2=1=\mathbf{1}+0$$

La partie décimale est nulle : stop ! Finalement : $(4,125)_{10}=(100,001)_2$

b) $(100,0101)_2=1\times 2^2+0\times 2^1+0\times 2^0+0\times 2^{-1}+1\times 2^{-2}+0\times 2^{-3}+1\times 2^{-4}=(4,3125)_{10}$

c)

$0,1\times 2=0,2=0+0,2$	ligne 1
$0,2\times 2=0,4=0+0,4$	ligne 2
$0,4\times 2=0,8=0+0,8$	ligne 3
$0,8\times 2=1,6=1+0,6$	ligne 4
$0,6\times 2=1,2=1+0,2$	ligne 5
$0,2\times 2=0,4=0+0,4$	On se retrouve comme à la ligne 2
$0,4\times 2=0,8=0+0,8$	Comme la ligne 3
$0,8\times 2=1,6=1+0,6$	Comme la ligne 4
$0,6\times 2=1,2=1+0,2$	Comme la ligne 5
$0,2\times 2=0,4=0+0,4$	On se retrouve comme à la ligne 2
$0,4\times 2=0,8=0+0,8$	Comme la ligne 3 ….......etc.

Le processus de "conversion" $(0,1)_{10}$ ne s'arrête pas, nous obtenons : "0,0001100110011...", le schéma "0011" se répète à "l'infini". $(0,1)_{10}$ ne peut être représenté EXACTEMENT !

d) Voir fichiers *Decimale-vString.py* et *Decimale-vTableau.py*
e) Voir fichiers *Decimale-vString.py* et *Decimale-vTableau.py*

Exercice 2 : codage en virgule flottante - norme IEEE-754

1.

a) $(10{,}59375)_{10}$

Le nombre est positif, donc le 1er bit sera un 0.

On convertit la partie entière et la partie décimale en base 2 :

- $(10)_{10} = (1010)_2$
- pour la partie décimale :

$$0{,}59375\times 2=1{,}1875=1+0{,}1875$$
$$0{,}1875\times 2=0{,}375=0+0{,}375$$
$$0{,}375\times 2=0{,}75=0+0{,}75$$
$$0{,}75\times 2=1{,}5=1+0{,}5$$
$$0{,}5\times 2=1=1+0$$

donc $(0{,}59375)_{10}=(0{,}10011)_2$

Donc : $(10{,}59375)_{10}=(1010{,}10011)_2=(1{,}01010011\times 2^{(3)_{10}})_2$ La partie fractionnaire de la mantisse sera donc 01010011.

Calcul de l'exposant n : n-127 = 3 donc n= $(3+127)_{10} = (130)_{10} = (10000010)_2$

Finalement, $(10{,}59375)_{10}$ se code en simple précision selon la norme IEEE-754 ainsi :

0 1000 0010 0101 0011 0000 0000 0000 000

2.

a) 1 0100 0110 1001 0000 0000 0000 0000 000

Le bit de signe est un 1, donc le nombre sera négatif.

Les 8 bits suivants (0100 0110) nous permettent de retrouver l'exposant n :

$n=2^6+2^2+2^1=70$ ddnc l'exposant réel est : $n-127=70-127=-57$

Les 23 bits restants (1001 0000 0000 0000 0000 000) correspondent à la partie fractionnaire de la mantisse. La mantisse est donc : **1**,1001

Enfin, on décale la virgule de 57 rangs à gauche :

Le nombre codé en base 10 est :

$(-0{,}000\ldots0011001)_2=-(2^{-57}+2^{-58}+2^{-61})\approx(-0{,}0000000000000000108420217)_{10}\approx(-1{,}08420217\times10^{-17})_{10}$

C'est approximatif !