

# Devoir Maison 1 : corrigé

## Exercice 1. Représentation des nombres

1. **Nombres entiers.** Pour la base 16, on utilise les chiffres 0, 1, ..., 9, A, B, C, ..., F.

| Base 10 | Base 2   | Base 16 |
|---------|----------|---------|
| 19      | 10011    | 13      |
| 170     | 10101010 | AA      |
| 171     | 10101010 | AB      |

2. **Nombres à virgule flottante.** On utilise les conventions données dans le **chapitre 2** : paragraphe 3.1.

On considère le nombre réel  $x = \frac{5}{6}$ . Approximations (par troncature)  $\tilde{x}$  en nombre à virgule flottante :

(a)  $N = 4$  chiffres significatifs et base  $b = 10$ .

$$\tilde{x}_1 = 8,333.10^{-1}$$

(b)  $N = 3$  chiffres significatifs et base  $b = 2$ .

$$\tilde{x}_2 = 1,10.10^{-1}$$

*Indication.* Pour (b) ; en base 10 :

$$\frac{5}{6} \times 2^3 = \frac{40}{6} = 6 + \frac{2}{3} = 13 + \varepsilon \text{ avec } 0 < \varepsilon < 1.$$

Donc, en base 2,  $x.10^3 = 110 + \varepsilon = 1,10.10^2 + \varepsilon$ .

Ainsi  $x = 1,10.10^{-1} + \varepsilon'$  avec  $0 < \varepsilon' < 10^{-3}$ .

3. Voici une série de commandes rédigées dans un interpréteur **Python**.

```
>>> a=1.0
>>> b=a+1.0
>>> while b-a==1.0 :
    a=a*2.0
    b=a+1.0
```

(a) Si les calculs étaient réalisés en valeurs exacts, que se passerait-il à l'exécution de la boucle **while** ?

**Réponse.**

Comme nous avons l'instruction  $b = a + 1.0$ , on peut imaginer que la condition  $b - a = 1.0$  est toujours vérifiée. **La boucle ne se termine pas.**

(b) Exécuter ces lignes de commandes sur votre ordinateur. Que constate-t'on ? Comment expliquer que la boucle se termine ?

On constate que la boucle se termine. L'affichage des dernières valeurs pour  $a$  et  $b$  donne :

$$a = b = 9007199254740992.0$$

Ainsi la boucle se termine car la condition  $a - b == 1.0$  n'est plus réalisée.

On remarque que  $9007199254740992.0 + 1.0 = 9007199254740992.0$ . Ces deux quantités sont d'un ordre de grandeur trop différent. Quand Python effectue la somme de ces deux nombres, l'arrondi réalisé pour respecter le nombre de chiffres significatifs imposés conduit à négliger le terme 1.0.

- (c) Le programme `nombreBoucles.py` affiche le nombre de tours effectués par la boucle `while`.

**Programme `nombreBoucles`.**

```
a=1.0
b=a+1.0
compteur = 0 # Compte les tours de boucle
while b-a==1.0 :
    a=a*2.0
    b=a+1.0
    compteur=compteur+1 # Le compteur s'incrémente après un tour de boucle
print(compteur)
```

## Exercice 2. Pour s'occuper pendant les longues soirées d'hiver

Vous pourrez trouver dans le cahier de texte une proposition de programme.