

## 2026 sujet 8

**Question 1**

```
def calcul_recettes():
    s = 0
    for i in range(500000):
        s = s + 2.27 + 5.19 + 1.81
    return s

print(calcul_recettes())
```

La sortie est :

4634999.999986519

Ce sont les erreurs d'arrondis qui s'accumulent.

**Question 2**

On utilise la fonction `int(nombre_binaire, 2)` pour convertir un nombre binaire en décimal.

```
def convertir_BCD_vers_decimal(liste_quartets):
    number = 0
    puissance = -2
    # On parcourt la liste à l'envers
    for i in range(len(liste_quartets) - 1, -1, -1):
        number = number + int(liste_quartets[i], 2) * 10**puissance
        puissance += 1
    return number

assert convertir_BCD_vers_decimal(['0001', '0011', '0101', '0110']) == 13.56
```

**Question 3**

`additionner_nombres_format_BCD('27', '35')` renvoie bien 62 mais quoq la forme 50 + 12, il y a donc bien une erreur.

```
>>> additionner_nombres_format_BCD('27', '35')
['0101', '1100', '0000', '0000']
```

```
def additionner_nombres_format_BCD(a, b):
    """
    Additionne deux nombres au format BCD, quartet par quartet.
    """
    liste_quartets1 = convertir_dec_vers_BCD(a)
    liste_quartets2 = convertir_dec_vers_BCD(b)
```

```

# Ajustement de la longueur
liste_quartets1, liste_quartets2 = aligner_quartets(liste_quartets1, liste_quartets2)

retenue = 0
resultat = []
longueur_max = max(len(liste_quartets1), len(liste_quartets2))

for i in range(longueur_max):
    index = longueur_max - i - 1

    # Addition binaire simple des quartets
    somme, retenue = additionner_binaire_quartets(liste_quartets1[index], liste_quartets2[index])

    # Correction
    somme, retenue = corriger_BCD(somme, retenue)

    resultat.insert(0, somme)

# Gestion de la dernière retenue éventuelle
if retenue == 1:
    resultat.insert(0, '0001')

return resultat

```

Le résultat est alors correct :

```

>>> additionner_nombres_format_BCD('27', '35')
['0110', '0010', '0000', '0000']

```

#### Question 4

Il y a une erreur `IndexError` :

```

>>> additionner_nombres_format_BCD('23', '4')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/home/thomas/Documents/work/enseignements/Informatique/Correction sujet bac/2020", line 1, in <module>
    somme, retenue = additionner_binaire_quartets(liste_quartets1[index], liste_quartets2[index])
IndexError: list index out of range

```

C'est parce que les listes ne font pas la même longueur. On modifie donc `aligner_quartets` :

```

def aligner_quartets(q1: list, q2: list) -> tuple:
    """
    Doit équilibrer les deux listes en ajoutant des '0000' à gauche
    de la liste la plus courte.
    """
    l1 = len(q1)
    l2 = len(q2)

    if l1 > l2:

```

```
    for i in range(l1 - l2):
        q2.insert(0, '0000')
if l2 > l1:
    for i in range(l2 - l1):
        q1.insert(0, '0000')
return q1, q2
```

qui semble fonctionner :

```
>>> aligner_quartets(convertir_dec_vers_BCD('4'), convertir_dec_vers_BCD('23'))
(['0000', '0100', '0000', '0000'], ['0010', '0011', '0000', '0000'])
```

On test à nouveau l'addition :

```
>>> additionner_nombres_format_BCD('23', '4')
['0010', '0111', '0000', '0000']
```

Cela fonctionne!