Corrigé sujet 20 - Année : 2022

Sujet 20 - 20222

1. Exercice 1

```python
def xor(a,b):
    resultat = []
    for i in range(len(a)):
        if a[i]==b[i]:
            resultat.append(0)
        else:
            resultat.append(1)
    return resultat
```

Commentaire

On peut aussi utiliser une définition de listes par compréhension.

2. Exercice 2

```
1   class Carre:
2       def __init__(self, tableau = [[]]):
3           self.ordre = len(tableau)
4           self.valeurs = tableau
5
6       def affiche(self):
7           '''Affiche un carré'''
8           for i in range(self.ordre):
9               print(self.valeurs[i])
10
11      def somme_ligne(self, i):
12          '''Calcule la somme des valeurs de la ligne i'''
13          return sum(self.valeurs[i])
14
15      def somme_col(self, j):
16          '''calcule la somme des valeurs de la colonne j'''
17          return sum([self.valeurs[i][j] for i in range(self.ordre)])
18
19  def est_magique(carre):
20      n = carre.ordre
21      s = carre.somme_ligne(0)
22
23      #test de la somme de chaque ligne
24      for i in range(1,n): #(1)
25          if carre.somme_ligne(i) != s:
26              return False
27
28      #test de la somme de chaque colonne
29      for j in range(n):
30          if carre.somme_col(j) != s: #(2)
31              return False
32
33      #test de la somme de chaque diagonale
34      if sum([carre.valeurs[k][k] for k in range(n)]) != s: #(3)
35              return False
36      if sum([carre.valeurs[k][n-1-k] for k in range(n)]) != s:
37              return False
38      return True   #(4)
```

1. Par la peine de tester la ligne d'indice 0, elle a servi à calculer la somme de référence `s` (ligne 21), on commence donc à 1.
2. On utilise la méthode `somme_col` de la classe `Carre`
3. La diagonale principale se caractérise par des indices de lignes et de colonne identiques.
4. Si on atteint cette ligne, tous les tests ont été passé avec succès, le carré est magique !

Attention

- Le code fourni utilise un objet mutable (une liste) comme paramètre par défaut d'une fonction :

  🐍 **Script Python**

  ```
  def __init__(self, tableau = [[]]):
  ```

  C'est une très mauvaise pratique car source d'erreurs, en effet la variable `tableau` étant mutable elle est modifiée par la fonction lors d'un premier appel et ne sera donc plus vide lors des appels suivants. Pour une solution à ce problème, on pourra par exemple consulter ce site