

2022 sujet 20

Exercice 1

```
def xor(tab1, tab2):
    res = []
    for i in range(len(tab1)):
        if tab1[i] == tab2[i]:
            res.append(0)
        else:
            res.append(1)
    return res

# Tests

a = [1,0,1,0,1,1,0,1]
b = [0,1,1,1,0,1,0,0]
c = [1,1,0,1]
d = [0,0,1,1]

assert(xor(a, b) == [1, 1, 0, 1, 1, 0, 0, 1])
assert(xor(c, d) == [1, 1, 1, 0])
```

Exercice 2

```
# Attention !!!
# Il ne faut jamais mettre un objet mutable comme valeur par défaut
# d'un paramètre de fonction car il est évalué une seule fois
# lorsque la fonction est définie.
# La classe Carre risque donc d'avoir un comportement erratique
# car plusieurs instances partageront le même tableau.

class Carre:
    def __init__(self, tableau = [[]]):
        self.ordre = len(tableau)
        self.valeurs = tableau

    def affiche(self):
        '''Affiche un carré'''
        for i in range(self.ordre):
            print(self.valeurs[i])
```

```

def somme_ligne(self, i):
    '''Calcule la somme des valeurs de la ligne i'''
    return sum(self.valeurs[i])

def somme_col(self, j):
    '''calcule la somme des valeurs de la colonne j'''
    return sum([self.valeurs[i][j] for i in range(self.ordre)])

def est_magique(carre):
    n = carre.ordre
    s = carre.somme_ligne(0)

    #test de la somme de chaque ligne
    for i in range(1, n):
        if carre.somme_ligne(i) != s:
            return False

    #test de la somme de chaque colonne
    for j in range(n):
        if carre.somme_col(j) != s:
            return False

    #test de la somme de chaque diagonale
    if sum([carre.valeurs[k][k] for k in range(n)]) != s:
        return False
    if sum([carre.valeurs[k][n-1-k] for k in range(n)]) != s:
        return False

    return s

# Tests

c2 = Carre([[1,1],[1,1]])
c3 = Carre([[2,9,4],[7,5,3],[6,1,8]])
c4 = Carre([[4,5,16,9],[14,7,2,11],[3,10,15,6],[13,12,8,1]])

assert(est_magique(c2) == 2)
assert(est_magique(c3) == 15)
assert(est_magique(c4) == False)

```