

Devoir surveillé n°2 – NSI
Correction

Exercice 1

```
class Carre:
    def __init__(self, liste, n):
        self.ordre = n
        self.tableau = [[liste[i + j * n] for i in range(n)]
                        for j in range(n)]

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

    def somme_ligne(self, i):
        '''Calcule la somme des valeurs de la ligne i'''
        somme = 0
        for j in range(self.ordre):
            somme = somme + self.tableau[i][j]
        return somme

    def somme_col(self, j):
        '''Calcule la somme des valeurs de la colonne j'''
        somme = 0
        for i in range(self.ordre):
            somme = somme + self.tableau[i][j]
        return somme

    def est_semimagique(self):
        s = self.somme_ligne(0)

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

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

c2 = Carre([1, 7, 7, 1], 2)
c3 = Carre([3, 4, 5, 4, 4, 4, 5, 4, 3], 3)
c3bis = Carre([2, 9, 4, 7, 0, 3, 6, 1, 8], 3)

assert c2.est_semimagique()
assert c3.est_semimagique()
assert not c3bis.est_semimagique()
```

Exercice 2

1. (a) `lait = Aliment(65.1, 3.32, 4.85, 3.63)`
(b) `lait.energie`
(c) `lait.proteines = 3.4`
2.

```
def energie_reelle(self,masse):  
    return masse*self.energie/100
```
3. (a) `nutrition['lait'].energie`
(b) `nutrition['lait'].energie_reelle(220)`
4.

```
def calcul_energie_tot(recette, nutrition):  
    """  
        recette est un dictionnaire qui contient la recette  
        nutrition est un dictionnaire qui contient des instances de la  
        classe Aliment  
    """  
    energie_tot = 0  
    for a,m in recette.items():  
        energie_tot = energie_tot + nutrition[a].energie_reelle(m)  
    return energie_tot
```