

## Corrigé DS 1 d'informatique

**Exercice 1**

On s'intéresse à la fonction suivante prenant en argument un nombre  $a$  de type `int`.

```
def fonction(a):
    if a > 10 :
        if a < 15 :
            return a + 10
        else :
            if a % 2 == 0:
                return a / 2
            else:
                return (a - 1) / 2
    else :
        if a % 2 == 0:
            return a + 1
        else:
            return a
```

1. Quelle valeur renvoie la fonction si l'utilisateur entre les valeurs suivantes : 13, 4, 18, 19, 5? Lorsque l'on teste la fonction avec la valeur 13 en entrée, la fonction renvoie 23.  
Lorsque l'on teste la fonction avec la valeur 4 en entrée, la fonction renvoie 5.  
Lorsque l'on teste la fonction avec la valeur 18 en entrée, la fonction renvoie 9.  
Lorsque l'on teste la fonction avec la valeur 19 en entrée, la fonction renvoie 9.  
Lorsque l'on teste la fonction avec la valeur 5 en entrée, la fonction renvoie 5.
2. La fonction peut elle renvoyer un nombre impaire plus grand que 30?  
Oui car si on appelle la fonction avec la valeur 62, la fonction renvoie 31.
3. La fonction peut elle renvoyer un nombre pair plus petit que 10?  
Oui car si on appelle la fonction avec la valeur 16, la fonction renvoie 8.
4. La fonction peut elle renvoyer le nombre 2?  
Non car
  - dans le cas où  $a > 10$  et  $a > 15$  la fonction renvoie un nombre compris strictement entre 20 et 25 (donc entre 21 et 24).
  - dans le cas où  $a > 10$  et  $a \leq 15$  la fonction renvoie un nombre supérieur ou égal à 7
  - Dans le cas où  $a \leq 10$ , le programme renvoie dans tous les cas un nombre impair.

Le nombre 2 ne peut donc jamais être renvoyé.

**Exercice 2**

Corrigé en classe

**Exercice 3**

```
def test(L, k):
    for x in L:
        if x == k:
            return True
    return False
```

**Exercice 4**

```
def multiplie(L, k):
    M = [] # création d'une nouvelle liste vide
    for i in range(len(L)):
        M.append(k * L[i])
    return M
```

**Exercice 5**

Ecrire une fonction qui prend en argument un entier  $n$  et qui renvoie la somme  $1^2 + 2^2 + 3^2 + \dots + n^2$ .

Attention, après un `def`, les deux points ainsi que l'indentation s'imposent!! Attention, si on veut faire une boucle `for` qui va jusqu'à  $n$ , on doit mettre `range(n+1)`

```
def somme_carre(n):  
    S=0  
    for i in range(1,n+1):  
        S=S+i**2  
    return S
```

**Exercice 6**

Ecrire une fonction qui prend en argument un entier  $n$  et qui renvoie le produit  $1^2 \times 2^2 \times 3^2 \times \dots \times n^2$ .

```
def produit_carre(n):  
    P=1  
    for i in range(1,n+1):  
        P=P*i**2  
    return P
```