

**N. S. I. 1<sup>ère</sup> - DS04 – Tableaux (à une dimension) (mercredi 13 janvier 2021)****Nom :****Prénom :****Classe :****PARTIE 1 L'itérateur `range`****Q1.** Quels affichages obtient-on lorsque l'on exécute le programme suivant ?

```
for i in range(5):  
    print(i)
```

**Q2.** Quels affichages obtient-on lorsque l'on exécute le programme suivant ?

```
for i in range(1, 5):  
    print(i**2)
```

**Q3.** Écrire une boucle `for` qui affiche les entiers de l'intervalle d'entiers  $\llbracket 0 ; 999 \rrbracket$  ?**Q4.** Écrire une boucle `for` qui affiche les entiers pairs de l'intervalle d'entiers  $\llbracket 0 ; 999 \rrbracket$  ? (non posé au DS)**Q5.** Écrire une boucle `for` qui affiche les entiers compris entre 10 et 20 inclus ?**PARTIE 2 Tableaux à une dimension**

On considère le tableau

```
T_ex = ['l', 'a', 'o', 'v', 'r', 'y', 's', 'd', 'g', 'n']
```

**Q6.** Quelle instruction donne le nombre d'éléments de `T_ex` ?**Q7.** Rappeler les numéros de positions positifs des lettres « a », « n », « r » dans le tableau `T_ex`.

```
T_ex = ['l', 'a', 'o', 'v', 'r', 'y', 's', 'd', 'g', 'n']
```

**Q8.** Que renvoient les instructions `T_ex[4]`, `T_ex[-4]`, `T_ex[10]` pour le tableau `T_ex` ?

**Q9.** Quelle est l'indice de position positif du premier élément d'un tableau `T` ?

**Q10.** Quelle est l'indice de position négatif de l'avant-dernier élément d'un tableau `T` ?

**Q11.** Quelle séquence d'instructions permet, à l'aide d'une boucle, d'afficher un à un, dans l'ordre, les éléments d'une liste `T` ? Par exemple, si `T = ['a', 'o', 'v', 'r']`, lorsque l'on exécute les instructions de la séquence, on doit obtenir l'affichage :

```
a
o
v
r
```

**Q12.** Proposer une (ou des) instructions pour construire le tableau `tab1` suivant, contenant 100 fois la valeur 1 : `tab1 = [1, 1, 1, 1, 1, 1, 1, 1, ..., 1, 1, 1]`.

**Q13.** Proposer une séquence d'instructions, utilisant une boucle, permettant de définir un tableau, `tab2`, de dix entiers tirés aléatoirement entre 1 et 10 inclus.

### PARTIE 3 Algorithmes sur les tableaux à une dimension

**Q14.** On considère la fonction `f` définie ci-dessous.

```
def f(T):
    for k in range(len(T)):
        print(T[k])
```

Que se passe-t-il lors de l'appel `f([4, 5, 6])` ?

On rappelle que :

- l'expression  $b // a$  calcule le **quotient** dans la division euclidienne d'un entier positif  $b$  par un entier strictement positif  $a$ .
- l'expression  $b \% a$  calcule le **reste** dans la division euclidienne d'un entier positif  $b$  par un entier strictement positif  $a$ .

Par exemple,  $13 // 5$  vaut 2 et  $13 \% 5$  vaut 3, car  $13 = 5 \times 2 + 3$ , ce que l'on obtient en posant l'opération :

$$\begin{array}{r} 13 \phantom{00} \\ 5 \overline{) 13} \\ \underline{10} \phantom{00} \\ 3 \phantom{00} \end{array}$$

**Q15.** Que vaut  $27 // 7$  ?

**Q16.** Que vaut  $27 \% 7$  ?

**Q17.** En fonction des réponses précédentes, compléter l'égalité  $27 = 7 \times \dots + \dots$ .

$n \% 2$  calcule le reste dans la division euclidienne par deux d'un entier  $n$ .

**Q18.** Quelles sont les valeurs possibles de  $n \% 2$  ?

On considère la fonction  $g$  définie ci-dessous :

```
def g(T):
    for k in range(len(T)):
        if T[k] % 2 == 0 :
            print(T[k])
```

**Q19.** Que se passe-t-il lors de l'appel  $g([4, 5, 6])$  ?

On rappelle que la fonction `somme` suivante calcule la somme des valeurs dans un tableau  $T$  :

```
def somme(T):
    s = 0
    for k in range(len(T)):
        s = s + T[k]
    return s
```

**Q20.** Que renvoie l'appel `somme([4, 5, 2, 3, 1])` ?

On modifie la fonction `somme` en une fonction `somme2`, de la façon suivante

```
def somme2(T):
    s = 0
    for k in range(len(T)):
        s = s + 2 * T[k]
    return s
```

**Q21.** Que renvoie l'appel `somme2([4, 5, 2, 3, 1])` ?

On modifie la fonction `somme` en une fonction `somme3`, de la façon suivante

```
def somme3(T):  
    s = 0  
    for k in range(len(T)):  
        if T[k] % 2 == 0:  
            s = s + T[k]  
    return s
```

Q22. Que renvoie l'appel `somme3([4, 5, 2, 3, 1])` ?

On considère la fonction `mystere1` suivante :

```
def mystere1(T, v):  
    for k in range(len(T)):  
        if T[k] == v :  
            return True  
    return False
```

Q23. Que renvoie l'appel `mystere1([4, 5, 2, 3, 1], 7)` ?

Q24. Que renvoie l'appel `mystere1([4, 5, 2, 3, 1], 3)` ?

Q25. En une phrase, expliquer ce que fait la fonction `mystere1`, c'est-à-dire en une phrase, expliquer ce que renvoie un appel `mystere1(T, v)` lorsque  $T$  est un tableau à une dimension et  $v$  une valeur.

On considère la fonction `maximum` suivante, prenant en argument un tableau  $T$ , non vide :

```
def maximum(T):  
    vmax = 0  
    for k in range(len(T)):  
        if T[k] > vmax :  
            vmax = T[k]  
    return vmax
```

Q26. Que renvoie l'appel `maximum([4, 5, 2, 3, 1])` ?

Q27. Que renvoie l'appel `maximum([-4, -5, -2, -3, -1])` ?

Q28. Corriger le code de la fonction `maximum` pour que l'appel `maximum([-4, -5, -2, -3, -1])` renvoie la même valeur que l'appel `max([-4, -5, -2, -3, -1])` où `max` est la fonction prédéfinie `max`, et de façon générale, pour tout tableau  $T$ , non vide, passé en argument.

Q29. Écrire une fonction `minimum(T)` qui renvoie le minimum d'un tableau  $T$ , non vide, comme le fait l'appel `min(T)` à la fonction prédéfinie `min(T)`.