

# Activité

## activité 2.1

Il est possible de visualiser le contenu d'un tuple ou d'un tableau directement dans la console.

Saisissez le code suivant dans la partie "éditeur" de spyder :

```
mon_tuple = (5, 8, 6, 9)
```

Exécutez le programme

Tapez ensuite *mon\_tuple* dans la partie "console"

## activité 2.2

Complétez le programme suivant afin que la variable *a* ait pour valeur 8

```
mon_tuple = (5, 8, 6, 9)
a = mon_tuple[...]
```

## activité 2.3

Complétez le programme suivant afin que la variable *a* ait pour valeur 8 et que la variable *b* ait pour valeur 5

```
a, b = (... , ...)
```

## activité 2.4

Complétez le programme ci-dessous pour qu'après exécution de ce dernier, le tableau *mon\_tab* soit composé des éléments suivants : [15, 8, 6, 9]

```
mon_tab = [5, 8, 6, 9]
mon_tab[...] = 15
```

## activité 2.5

Soit le tableau suivant :

```
tab = [3,3,6,9]
```

Quelle sera la composition de ce tableau si on exécute la ligne suivante :

```
tab.append(0)
```

## activité 2.6

```
mon_tab = [1, 2, 3, 4]  
del mon_tab[1]
```

Donnez la composition du tableau *mon\_tab* après l'exécution du programme ci-dessus

## activité 2.7

```
mon_tab = [5, 8, 6, 9, 15, 0]  
a = len(mon_tab)
```

Quelle est la valeur de la variable *a* après l'exécution du programme ci-dessus ?

## activité 2.8

```
tab = [1, 2, 3]  
s = 0  
for t in tab:  
    s = s + t
```

Quelle est la valeur de la variable *s* après l'exécution du programme ci-dessus. Vérifiez votre réponse à l'aide de la console.

## activité 2.9

```
tab = [5, 3, 4, 8]  
mon_tab = [2*t for t in tab if t > 4]
```

Quelle est la composition du tableau *mon\_tab* après l'exécution du programme ci-dessus. Vérifiez votre réponse à l'aide de la console.

## activité 2.10

Soit le programme suivant :

```
m = [[1, 3, 4],
```

```
[5, 6, 8],  
[2, 1, 3],  
[7, 8, 15]]  
a = m[0][1]
```

Quelle est la valeur de la variable *a* après l'exécution de ce programme ? Vérifiez votre réponse à l'aide de la console.

## activité 2.11

```
m = [[1, 3],  
     [5, 8],  
     [2, 3]]  
nb_colonne = 2  
nb_ligne = 2  
a = 0  
for i in range(0, nb_ligne):  
    for j in range(0, nb_colonne):  
        a = a + m[i][j]
```

Quelle est la valeur de la variable *a* après l'exécution de ce programme ? Vérifiez votre réponse à l'aide de la console.

## activité 2.12

La fonction *recherche\_max* prend en paramètre un tableau et renvoie la plus grande valeur présente dans le tableau (le tableau est constitué d'entiers positifs ou nuls).

Par exemple, *recherche\_max([4, 3, 0, 5])* renvoie 5

```
def recherche_max(tab):  
    maxi = ...  
    for t in tab :  
        if ... > maxi :  
            maxi = ...  
    return ...
```

Compétez la fonction ci-dessus

## activité 2.13

La fonction *somme* prend en paramètres un tableau *tab* contenant des entiers. Cette fonction renvoie la somme des entiers présents dans le tableau *tab*.

Exemples :

```
>>> somme([3,5,8,4])
20
```

```
def somme(tab):
    s = ...
    for t in ... :
        s = s + ...
    return ...
```

Complétez la fonction *somme* ci-dessus.

## activité 2.14

La fonction *recherche* prend en paramètres un tableau *tab* contenant des entiers et un entier *n*. Cette fonction renvoie l'indice de position de l'entier *n* s'il est présent dans le tableau *tab* ou -1 dans le cas contraire (on partira du principe que le tableau ne peut pas contenir 2 fois le même entier).

Exemples :

```
>>> recherche([3,5,8,34], 8)
2
>>> recherche([3,5,8,34], 42)
-1
```

```
def recherche(tab, n):
    indice = ...
    i = 0
    for t in ... :
        if n == ... :
            indice = ...
        i = i + ...
    return ...
```

Complétez la fonction *recherche* ci-dessus.

## activité 2.15

Écrire une fonction *moyenne* qui prend en paramètre un tableau d'entiers non vide *tab* et qui renvoie la moyenne de ces entiers.