

Exercice 1 :

On se donne la fonction suivante :

```
def echanger(a, b):
    z=a
    a=b
    b=z
a=5
b=3
echanger(a, b)
print ("a_vaut:", a)
print ("b_vaut:", b)
```

- (1) À quoi sert cette fonction ?
- (2) Tester la fonction, que remarquez-vous ?
- (3) Corriger ce programme pour le faire fonctionner.

Exercice 2 :

On se donne la fonction suivante :

```
def echanger2(l1, l2):
    z=l1[0]
    l1[0]=l2[0]
    l2[0]=z
l1 = [5]
l2 = [3]
echanger2(l1, l2)
print ("l1_vaut:", l1)
print ("l2_vaut:", l2)
```

- (1) Tester ce programme.
- (2) Quelle est la différence avec le programme précédent ?

Exercice 3 :

Tester la fonction :

```
def echanger3():
    global a, b
    z=a
    a=b
    b=z
```

Expliquer pourquoi ce programme fonctionne.

Exercice 4 :

Qu'affiche le programme suivant ? Pourquoi ?

```
def g(x):
    global a
    a=10
    return 2*x
def f(x):
    v=1
    return g(x)+v
a=3
print (f(a)+a)
```

Exercice 5 :

$n!$ est le nombre définie par :

$$n! = 1 \times 2 \times 3 \times \dots \times n.$$

Ce nombre est le terme général de la suite définie par :

$$u_{n+1} = (n+1)u_n \text{ et } u_0 = 1$$

On se donne la fonction suivante :

```
def factorielle(n):
    if n == 0:
        return 1
    else:
        return n * factorielle(n-1)
```

- (1) Tester cette fonction pour différentes valeurs de n .
- (2) Pourquoi ce programme marche-t-il ?

Exercice 6 :

Écrire une fonction **chercher** qui prend en argument une liste et un nombre et renvoie l'indice de ce nombre dans la liste si il existe et -1 sinon. Par exemple :

```
L = [1,2,5]
print (chercher(L,2))
--> 1
print (chercher(L,3))
--> -1
```