

Réversivité : Exercices en mode " débranché" (sans ordinateur !)

Exercice 1 :

1. On donne la fonction f définie ci-dessous par :

```
[ ]: def f(n):  
    return n * f(n - 1)
```

Que se passe-t-il si on appelle f(3) ?

2. On donne la fonction g définie ci-dessous par :

```
[ ]: def g(n):  
    if n == 0:  
        return 1  
    else :  
        return n * g(n - 1)
```

Que se passe-t-il si on appelle g(3) ?

3. On donne la fonction h définie ci-dessous par :

```
[ ]: def h(n):  
    if n == 0:  
        return 1  
    else :  
        return n * h(n) - 1
```

Que se passe-t-il si on appelle f(3) ?

4. On donne la fonction p définie ci-dessous par :

```
[ ]: def p(n):  
    if n == 0:  
        return 1  
    else :  
        return n * p(n - 2)
```

Que se passe-t-il si on appelle p(3) ?

Exercice 2 : soit la fonction récursive f ci-dessous :

```
def f(a,b):  
    """ a et b sont des  
        entiers naturels non nuls  
    """  
    if b==1:  
        return a  
    else:  
        return a+f(a,b-1)
```

1. Quel est le résultat affiché par ce programme ?
2. Quel est le cas de base de cette fonction récursive ?
.....
3. Qu'est ce qui garantit dans les appels récursifs que le
programme va finir par s'arrêter ?
.....
.....
.....
4. Que fait cette fonction ?
.....

Exercice 3 :

1. Quelles sont les affichages produits par ces deux fonctions avec n=5
2. Vérifier vos résultats dans une console

```
def fonction(n):  
    if n>0:  
        print(n)  
        fonction(n-1)  
  
def fonction_bis(n):  
    if n>0:  
        fonction_bis(n-1)  
        print(n)
```