

Exercices sur la récursivité - Terminale NSI

Exercice 1 : Compléter du code à trous

Complétez le code suivant pour obtenir une fonction récursive qui calcule la somme des nombres de 1 à n :

```
def somme(n):  
    if n == 0:  
        return ____ # cas de base  
    else:  
        return ____ + somme(____) # cas récursif
```

Exercice 2 : Écrire une fonction récursive à partir d'un énoncé

Énoncé : On souhaite écrire une fonction récursive `inverse(mot)` qui retourne un mot inversé **sans utiliser de boucle ni de slicing**.

Exemple :\

- `inverse("nsi")` → "isn"
1. Quel est le **cas de base** ?\
 2. Quel est le **cas récursif** ?\
 3. Écrivez la fonction complète.

Exercice 3 : Récursivité et affichage

Que va afficher le programme suivant ?

```
def mystere(n):  
    if n == 0:  
        print("Stop")  
    else:  
        print(n)  
        mystere(n - 1)  
  
mystere(3)
```

1. Donnez la sortie complète du programme.\
2. Quel est le rôle de cette fonction ?

Exercice 4 : Écriture libre

Énoncé : Écrire une fonction récursive `puissance(x, n)` qui calcule x^n (x puissance n).

- Quel est le **cas de base** ?\
 - Quel est le **cas récursif** ?\
 - Écrivez la fonction en Python.
-

Exercice 5 : Analyse d'un code récursif

On considère la fonction suivante :

```
def f(n):  
    if n <= 1:  
        return 1  
    else:  
        return f(n-1) + f(n-2)
```

1. Quel est le **cas de base** ?\
 2. Que calcule cette fonction ?\
 3. Calculez `f(5)` en détaillant les appels récursifs.
-