

# Travaux Dirigés : Preuve de Terminaison et Correction des Programmes Python

Enseignant : Azzouz MAZZA

Email : mazza8azzouz@gmail.com

Matière : Informatique.

Niveau : MPSI.

Etablissement : CPGE Oujda.

## 🎯 Objectif

Appliquer les concepts d'invariants et de variants de boucle pour prouver la terminaison et la correction des programmes Python.

## Exercices

### Exercice 1: Somme des entiers de 1 à $n$

#### Programme :

```
def somme_n(n):  
    s = 0  
    i = 1  
    while i <= n:  
        s += i  
        i += 1  
    return s
```

#### Questions :

1. Définissez un invariant de boucle pour ce programme.
2. Prouvez que la boucle termine à l'aide d'un variant.
3. Montrez que la fonction retourne bien la somme des entiers de 1 à  $n$ .

### Exercice 2: Recherche séquentielle

#### Programme :

```
def recherche_sequentielle(T, x):  
    i = 0  
    while i < len(T):  
        if T[i] == x:  
            return i  
        i += 1  
    return -1
```

#### Questions :

1. Définissez un invariant de boucle pour ce programme.
2. Prouvez que la boucle termine à l'aide d'un variant.
3. Montrez que la fonction retourne correctement l'indice de  $x$  si  $x$  est présent dans  $T$ , ou  $-1$  sinon.

**Exercice 3: Calcul du PGCD****Programme :**

```
def pgcd(a, b):  
    while b != 0:  
        a, b = b, a % b  
    return a
```

**Questions :**

1. Définissez un invariant de boucle pour ce programme.
2. Prouvez que la boucle termine à l'aide d'un variant.
3. Montrez que la fonction retourne bien le PGCD de  $a$  et  $b$ .

**Exercice 4: Inversion de chaîne****Programme :**

```
def inverse_chaine(s):  
    return s[::-1]
```

**Questions :**

1. Montrez que la fonction retourne bien la chaîne inversée.
2. Réécrivez la fonction sans utiliser l'indexation avancée `[::-1]` et prouvez la terminaison et la correction.

**Exercice 5: Factorielle d'un entier****Programme :**

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

**Questions :**

1. Définissez un invariant de boucle pour ce programme, et prouvez sa correction.
2. Prouvez que l'appel récursif se termine.