UNIVERSITÉ DE DOUALA
ÉCOLE SUPÉRIEURE TECHNIQUE LA SALLE
Algorithmique

Année académique 2026-2027
Classe : **IE4**
Pr Moffo Lonla Bertrand

Travaux Dirigés n° 4

Tableaux, structures, fonctions, procédures et pointeurs

Dans toute la fiche on utilise les déclarations globales : `Constantes NMAX = 100` et `Types t_tab = tableau[1..NMAX] d'entier`.

Exercice 1 : Fonction de recherche

Écrire la fonction `Fonction Occurrences(T : t_tab, n : entier, x : entier) : entier` qui retourne le nombre de fois que `x` apparaît dans les `n` premières cases de `T`. Quelle est sa complexité ?

Exercice 2 : Procédures sur un tableau

1. Écrire la procédure `Saisir(var T : t_tab, var n : entier)` qui lit une taille n contrôlée ($1 \leq n \leq$ NMAX) puis les n éléments.
2. Écrire la procédure `Inverser(var T : t_tab, n : entier)` qui renverse l'ordre des éléments (`[3, 8, 1, 6]` devient `[6, 1, 8, 3]`).
3. Écrire l'algorithme principal qui saisit un tableau, l'inverse et l'affiche.
4. Pourquoi `T` est-il passé par référence dans `Inverser`, mais pas `n` ?

Exercice 3 : Brin d'ADN complémentaire

Un brin d'ADN est une suite de bases codées `'A'`, `'C'`, `'G'`, `'T'`. Dans le brin complémentaire, `'A'` et `'T'` se correspondent, ainsi que `'C'` et `'G'`.

1. Écrire la fonction `Complement(b : caractère) : caractère`.
2. Écrire un algorithme qui lit les bases d'un brin (saisie contrôlée, fin de saisie marquée par `'X'`, 100 bases au plus), puis affiche le brin complémentaire.

Exercice 4 : Structures : gestion de notes

On définit la structure `Etudiant` (champs : `nom` chaîne, `note` réel) et le type `t_classe = tableau[1..60] d'Etudiant`.

1. Écrire la fonction `Moyenne(C : t_classe, n : entier) : réel`.
2. Écrire la procédure `AfficherAdmis(C : t_classe, n : entier)` qui affiche le nom des étudiants ayant au moins 10, puis leur nombre.

Exercice 5 : Passage de paramètres et pointeurs : simulation

Simuler l'algorithme suivant et donner l'affichage.

```
1 Procédure MODIF(a : entier, var b : entier, p : ↑entier)
2     Début
3         a ← a + 10
4         b ← b + 10
5         p↑ ← p↑ + 10
6     Fin
7 FinProcédure

  Algorithme : TESTPARAM
  Variables
```

```
    x, y, z : entier
Début
    x ← 1
    y ← 2
    z ← 3
    MODIF(x, y, &z)
    Écrire(x, " ", y, " ", z)
Fin
```

Exercice 6 : Récursivité

1. Écrire une fonction récursive `Puissance(x : réel, n : entier) : réel` calculant x^n pour $n \geq 0$.
2. Écrire une fonction récursive `PGCD(a, b : entier) : entier` fondée sur $\text{pgcd}(a, b) = \text{pgcd}(b, a \bmod b)$ et $\text{pgcd}(a, 0) = a$. Dérouler `PGCD(84, 36)`.