

# ■ Cheatsheet Algorithmique (Pseudo-code)

## 1■■■ TYPES DE DONNÉES

- ENTIER → nombre sans partie décimale
- RÉEL → nombre à virgule (utile pour moyennes, mesures, divisions)
- CHAÎNE → texte
- BOOLÉEN → vrai ou faux (logique binaire)
- TABLEAU(T) → structure indexée contenant des éléments de type T
- STRUCTURE X → regroupe plusieurs variables sous un même nom (ex : Apprenant = nom, prénom, notes)

## 2■■■ MOTS-CLÉS ET SYNTAXE

- VAR nom : Type ← valeur → déclare une variable et lui donne une valeur initiale
- CONST → variable constante (non modifiable)
- DEBUT / FIN → délimitent un bloc d'instructions
- FONCTION nom(params) : Type → bloc réutilisable qui calcule et RETOURNE une valeur
- PROCÉDURE nom(params) → bloc réutilisable sans valeur de retour
- RETOURNE → renvoie une valeur à la fin d'une fonction
- POUR ... FAIRE / POUR CHAQUE ... DANS ... FAIRE / FIN POUR → boucles
- SI ... ALORS / SINON / FIN SI → conditions
- TANT QUE ... FAIRE / FIN TANT QUE → boucle conditionnelle
- CET → référence à l'objet ou la structure actuelle (équivalent de "this")
- LEN(tableau) → donne la taille (nombre d'éléments) d'un tableau
- IS → vérifie le type d'un élément (ex : "SI x IS Tableau")

## 3■■■ BONNES PRATIQUES

- Une fonction = une seule responsabilité
- Ne pas mélanger affichage et calcul dans une même fonction
- Toujours initialiser les variables avant utilisation
- Utiliser des noms explicites (ex : calculerMoyenne, afficherResultats)
- Tester chaque fonction indépendamment
- Éviter les effets de bord (modifier des variables globales depuis une fonction)
- Toujours protéger les divisions : vérifier que le dénominateur  $\neq 0$

## 4■■■ RÉCURSIVITÉ

- Une fonction récursive s'appelle elle-même.
- Elle doit avoir une **\*\*condition d'arrêt\*\*** pour éviter la boucle infinie.
- Exemple : calcul d'une factorielle.

Exemple :

```
FONCTION factorielle(n : ENTIER) : ENTIER
SI n = 0 ALORS RETOURNE 1
SINON RETOURNE n * factorielle(n - 1)
FIN FONCTION
```

## 5■■■ COMPLEXITÉ ALGORITHMIQUE (Big O)

Opération	Complexité moyenne
Accès à un élément (tableau)	$O(1)$
Parcours de n éléments	$O(n)$
Tri basique (insertion, sélection)	$O(n^2)$
Tri efficace (merge, quick)	$O(n \log n)$
Recherche dichotomique	$O(\log n)$

## 6■■■ EXEMPLE COMPLET (calcul de moyenne de classe)

```
STRUCTURE Apprenant
VAR nom : CHAINE
VAR prenom : CHAINE
VAR notes : Tableau(REEL)
FIN STRUCTURE
```

```
FONCTION calculerMoyenneClasse(classe : Tableau(Apprenant)) : REEL
VAR somme ← 0
VAR compteur ← 0
POUR CHAQUE etudiant DANS classe FAIRE
POUR CHAQUE note DANS etudiant.notes FAIRE
somme ← somme + note
compteur ← compteur + 1
FIN POUR
FIN POUR
RETOURNE somme / compteur
FIN FONCTION
```