Bienvenue à ProSkills IT – Formations professionnelles au Togo

Fiche du cours 50 h Titre: DSA101 - Structures de données & Algorithmes 1 **Description:** Démarrage concret des bases incontournables en algorithmique. On apprend à raisonner en complexité (Big-O) et à manipuler les structures fondamentales (tableaux/listes, listes chaînées, piles, files, tables de hachage) et les tris de base. Exemples en Python avec parallèles C++/Java, et Csharp: les concepts restent valables dans tout langage. Prérequis : savoir écrire des programmes simples (variables, conditions, boucles). Objectifs: Expliquer Big-O (temps/espace), meilleure/moyenne/pire complexité et coût amorti (intuition).* Implémenter et utiliser tableaux/listes, listes chaînées, piles (LIFO), files (FIFO), deques.* Concevoir une table de hachage (hash, collisions : chaînage / open addressing, facteur de charge).* Programmer et comparer les tris de base (insertion, sélection, bulle) et la recherche (linéaire/binaire).* Choisir la structure adaptée à un problème, justifier par la complexité. Chapitres:

- 1. Complexité & Big-O (notations, intuition, amorti)*
- 2. Tableaux & listes (opérations, parcours, recherche)*
- 3. Listes chaînées (simple/double): insertion/suppression*
- 4. Piles: implémentations & cas d'usage (undo, parenthèses)*
- 5. Files & deques : file circulaire, files de tâches*
- 6. Hash maps/sets: collisions, redimensionnement, perfs*
- 7. Recherche (linéaire/binaire) & tris de base (comparatif, stabilité)*
- 8. Atelier problèmes (stacks/queues/hash) exercices guidés*
- 9. Atelier problèmes (recherche/tri) mini-bench & analyse*
- Mini-projet DSA-101 (au choix : analyse de logs, compteur de fréquences, validateur d'expressions)

À la fin :

Vous saurez analyser la complexité, implémenter les structures de base (listes, piles, files, hash maps), choisir un tri simple, et résoudre des problèmes concrets en transposant les idées entre Python, C++ et Java.