

# **Rapport de Projet**

## **Mini-Projet 2 - Gestion d'une Bibliothèque**

# Introduction

Projet consiste à implémenter une gestion de bibliothèque avec deux structures de données, une liste chaînée et une table de hachage.

**L'objectif** est d'apprendre à comparer ces deux structures en termes de performance, notamment dans le cadre de la recherche de livres.

La bibliothèque contient des livres, et chaque livre est représenté par un numéro d'enregistrement, un titre, et un auteur. on a implémenté deux solutions pour gérer ces données

- 1) utilisant une liste chaînée (Partie 1)
- 2) utilisant une table de hachage (Partie 2).

- **Objectifs et Structures de Données**

- **Liste Chaînée**

- Une bibliothèque stockant des livres sous forme de structures chaînées.

- **Table de Hachage**

- Une bibliothèque utilisant une table de hachage avec chaînage pour résoudre les collisions.

- **Fonctions Principales:**

- Liste Chaînée Insertion en tête, recherche par numéro ou titre, suppression, fusion.
    - Table de Hachage Insertion avec fonction de hachage, recherche par auteur.
    - Fichiers Chargement et sauvegarde de la bibliothèque.
    - Interface menu interactif permettant les opérations de gestion.

# Algorithmes Clés

- Insertion:
  - Liste chaînée on ajout en début de liste
  - Table de hachage on calcul de la clé et insert à l'index correspondant.
- Recherche.
  - Dans liste chaînée c'est parcours séquentiel.
  - Table de hachage c'est accès direct via la clé de hachage
- Suppression :
  - Suppression en adaptant les chaînages.

# Comparaison des Performances

- Recherche :
  - Liste chaînée :  $O(n)$ , car il faut parcourir toute la liste.
  - Table de hachage :  $O(1)$  en moyenne,  $O(n)$  en cas de collisions.
- Gestion des exemplaires multiples :
  - Liste chaînée :  $O(n^2)$  en comparant tous les livres
  - Table de hachage :  $O(n)$  dans le pire des cas.

# Conclusion

- Table de hachage recommande pour une gestion rapide et efficace des livres.
- Liste chaînée utile pour des implémentations plus simples.
- En conclusion, la table de hachage offre de meilleures performances pour des bibliothèques volumineuses.