

Algorithmes et structures de données : Travail 3 — Structures de données linéaires

Ce travail compte pour 4% de la session et porte sur les structures de données linéaires. Le code source pour le travail est disponible à <https://github.com/jfim/algo-rithmes-travail-3>. Vous ne pouvez pas utiliser les fonctions de la librairie standard. Vous devez remettre votre fichier `travail3.cpp` par LÉA avant le début du cours du 14 avril. La version papier du document doit être remise au début du cours du 14 avril.

Nom : _____

1. (12 points) Comme vu dans le cours, un vecteur est une structure linéaire redimensionnable qui possède les mêmes caractéristiques de performance qu'un tableau. Une liste chaînée est aussi une structure linéaire, mais avec des caractéristiques de performance différentes.
 - (a) Implementez un vecteur. La fonction `AjouteElement` prend en paramètre un index. S'il vaut -1, vous devez ajouter l'élément à la fin du vecteur. Sinon, vous devez l'insérer au bon index. Par exemple, si le vecteur contient les valeurs 1, 3 et 5, puis qu'on insère la valeur 8 à l'index 2, il contiendra les valeurs 1, 3, 8 et 5 (le premier élément possède l'index 0).
 - (b) Implementez une liste chaînée. L'interface est la même que celle du vecteur.
2. (2 points) Dans l'évaluation de votre liste chaînée, l'on supprime le premier million d'éléments de la liste chaînée. Selon vous, est-ce que cette opération est plus rapide avec un vecteur ? Expliquez pourquoi. ☐ Oui ☐ Non

3. (2 points) Votre liste chaînée possède un pointeur vers le dernier nœud. Selon vous, si la liste chaînée ne garde pas de pointeur vers le dernier nœud, est-ce que la performance de la liste chaînée sera affectée ? Expliquez pourquoi.

4. (2 points) La liste chaînée et le vecteur possèdent les deux un temps d'exécution de $O(1)$ pour l'insertion à la fin de la structure de données. Cependant, le vecteur prend moins de temps pour insérer 25 millions d'éléments que la liste chaînée. Ne devraient-ils pas avoir la même performance ? Expliquez votre raisonnement.

5. Selon vous, entre la liste chaînée et le vecteur :
- (a) (1 point) Laquelle consomme le moins de mémoire ? ☐ Liste chaînée ☐ Vecteur
 - (b) (1 point) Laquelle est la plus rapide pour trouver si une valeur existe ? ☐ Liste chaînée ☐ Vecteur