

POLYNOMES - Durée : 3 séances - Dossier à rendre pour la semaine du 06/03/2023.

On désire créer une bibliothèque de fonctions pour gérer un ensemble de polynômes à une variable, de degré pouvant être très grand mais contenant de nombreux coefficients nuls. Ces polynômes sont implantés en mémoire dans des listes chaînées où chaque cellule est une structure qui contient un terme d'un polynôme. Les cellules sont triées par degrés croissants. On ne peut pas trouver 2 termes de même degré dans un polynôme. Les termes nuls ne sont pas représentés. Un polynôme nul est donc représenté par une liste vide. Chaque liste est accédée au moyen d'un pointeur de tête.

Chaque polynôme à traiter est stocké sur disque dans un fichier texte. Chaque ligne du fichier contient un coefficient réel (sous forme décimale ou scientifique) et un degré entier positif ou nul. **Les fichiers sont triés et sans erreur.**

- 1) Écrire une fonction de création et une fonction d'insertion de cellule. Écrire une procédure qui utilise les fonctions précédentes pour créer un polynôme à partir d'un fichier. Écrire une procédure d'affichage de polynôme. Écrire les fonctions de recherche (pour mise à jour) et de suppression de cellule. **TOUTES LES FONCTIONS DEVRONT UTILISER EXACTEMENT LES ALGORITHMES ECRITS EN COURS (PRECEDENT EN POINTEUR DE POINTEUR).**
- 2) Écrire une fonction de dérivation d'un polynôme (la dérivée remplace le polynôme original).
- 3) Écrire la fonction d'addition de 2 polynômes P1 et P2. La somme est stockée dans P1, P2 disparaît.
- 4) Écrire la fonction qui calcule le produit de 2 polynômes P1 et P2, le produit est stocké dans un nouveau polynôme P3, créé par allocations de cellules. P3 est conforme aux règles énoncées au paragraphe 1.
- 5) Bonus : modifier les fonctions précédentes de manière à sauvegarder chaque résultat dans un fichier au lieu de l'afficher.

Remarque importante : Ce TP sera réalisé en utilisant la méthode d'unitest vue en TP de C. L'organisation du code source est fixée. Télécharger le squelette du programme avant de commencer la programmation et veiller à ajouter les fonctions utilisateurs dans le bon fichier.

CONDITION D'ACCES AUX MACHINES

Chaque séance de TP doit être préparée, les algorithmes doivent être écrits (ou retrouves dans les TD).

TRAVAIL A FAIRE :

1ère séance : question 1 (tests compris). 2ème séance : questions 2 et 3. 3ème séance : questions 4 et 5.

REDIGER LE DOSSIER DE PROGRAMMATION CORRESPONDANT :

Présentation générale :

- description de l'objet du TP (3 lignes),
- description et schéma de la structure de données et des fichiers de données utilisés,
- organisation du code source (dans quels fichiers sources sont les fonctions).

Détail de chaque fonction (programme principal compris) :

- algorithme de principe,
- lexique des notations (paramètres et variables locales) en en-tête.
- code source commenté (respecter le guide de style),

Compte rendu d'exécution

- un **fichier zip** nommé sous la forme **vous_Nom_Prenom.zip**, permettant de créer automatiquement un répertoire nommé **Votre_Nom_Prenom** après l'extraction.
- Le dossier doit contenir tous les codes source (**pas les .o, ni les exécutables**), un makefile, et les jeux de test complets (chaque fichier entrée et les résultats correspondants), tous les cas particuliers seront testés (en faire la liste).

Le TP est noté sur le contenu du dossier, la **qualité du code** et l'organisation et la **complétude** des jeux d'essais.