

PROJET Polynômes

OBSERVATIONS LIMINAIRES

- Pour la bonne réalisation de ce projet, la lecture des 2 documents «*Art de la Programmation en Langage C*» et «*Méthodologie de développement de projet en Langage C*» est requise.
- En résumé, une programmation modulaire, des tests et des commentaires sont indispensables.
- Par ailleurs, un rapport est demandé. Le document «*Rédaction d'un rapport*» sera consulté avec profit.

MODELISATIONS ET FONCTIONS SUR LES POLYNOMES

- Diverses fonctionnalités sur les polynômes au travers de diverses modélisations sont demandées.
- Seuls les polynômes à variable réelle et à coefficients entiers sont considérés.
- Les polynômes sont toujours réduits et ordonnés par ordre croissant du degré de leurs monômes.
- **Les fonctions sur les polynômes utilisent les fonctions sur les monômes développées en amont.**

I INTRODUCTION

- Le programme doit fournir des fonctions de tests sur les monômes et sur les fonctionnalités demandées sur les polynômes.
- Valgrind ne doit détecter aucune fuite mémoire (utilisation de valgrind).

La fonction *main* affichera un menu permettant de s'aiguiller sur les différents choix possibles :

- tests pour chacune des modélisations demandées et leurs fonctionnalités demandées
- application pour chacune des modélisations demandées et leurs fonctionnalités demandées

II CAHIER DES CHARGES GENERAL: FONCTIONNALITES F1 F2 ... Fn

- F1 **lire** un polynôme initialisé en dur selon l'ordre croissant de leur monômes
- F2 **lire** un polynôme dans un fichier
Le **format de lecture** d'un polynôme est illustré ci-dessous pour le polynôme $-4 + 2.5 x^5 - 4.7 x^{11}$, le nombre de monômes est donné en première ligne du fichier.
3
4 0 2.5 5 -4.7 11
- F3 **écrire** un polynôme sur la sortie standard
Le polynôme $-3 + 2.5 x^5 - 4.7 x^{11}$ s'écrit sous cette forme.
- F4 **écrire** un polynôme dans un fichier
Le **format d'écriture** d'un polynôme est spécifié par l'exemple suivant :
Le polynôme $-3 + 2.5 x^5 - 4.7 x^{11}$ s'écrit ainsi :
3
-3 0 2.5 5 -4.7 11
3 est le nombre de monômes et -3 0 le premier monôme de coefficient -3 et de degré 0, 2.5 5 le deuxième monôme de coefficient 2.5 et de degré 5
- F5 **calculer la somme de deux polynômes***
- F6 **calculer la produit de deux polynômes***
- F7 **calculer la valeur d'un polynôme pour une valeur entière de x**
 - selon la méthode la plus évidente...
 - selon la **méthode de Horner** (optionnel, voir Annexe)
- F8 **calculer la dérivée d'un polynôme***
- F9 **calculer la dérivée de la somme de deux polynômes***
rappel : $(P(x) + Q(x))' = P(x)' + Q(x)'$
- F10 **calculer la dérivée du produit de deux polynômes***
rappel : $(P(x) \times Q(x))' = P(x)' \times Q(x) + P(x) \times Q'(x)$

* Toutes les fonctions ont des polynômes réduits et ordonnés par degré croissant de leurs monômes en paramètres et aussi en retour si besoin est. L'algorithme de tri choisi est libre, il sera explicité dans le rapport.

III MODELISATION DES POLYNOMES ET FONCTIONNALITES DEMANDEES

MOD1 : Modélisation avec un tableau statique de monômes alloués statiquement

Un polynôme est représenté par un tableau statique de monômes, son nombre de monômes et son degré (degré du monôme le plus élevé).

Un monôme est représenté par son coefficient réel et son degré entier.

Les tableaux statiques ont une taille physique constante de 10. Il est important de gérer leur taille logique.

Fonctionnalités demandées : F1 F2 F3 F4 F5 F6 F7 F8 F9 F10

MOD2 : Modélisation avec un tableau dynamique de monômes alloués dynamiquement

Un polynôme est représenté par un tableau dynamique de monômes, son nombre de monômes et son degré (degré du monôme le plus élevé).

Un monôme est représenté par son coefficient réel et son degré entier.

Fonctionnalités demandées : F2 F3 F5 F8 F9

MOD3: Modélisation avec une liste chaînée de monômes

Le chaînage est réalisé via des maillons (cellules) contenant nécessairement un champ modélisant un monôme alloué statiquement

La modélisation d'un monôme est celle des modélisations précédentes.

Fonctionnalités demandées : F2 F3 F5 F8 F9

Certaines fonctions pourront être récursives.

MOD4: Modélisation avec une liste chaînée de monômes alloués dynamiquement

Le chaînage est réalisé via des maillons (cellules) contenant nécessairement un champ modélisant un monôme alloué dynamiquement.

La modélisation d'un monôme est celle des modélisations précédentes.

Fonctionnalités demandées : F2 F3 F5

Certaines fonctions pourront être récursives.

Indications :

Il faut bien réfléchir aux algorithmes permettant la réalisation de ces fonctionnalités notamment ceux sur les listes chaînées.

L'ajout ou la suppression de monômes lors de la réalisation de certaines fonctionnalités sont à schématiser dans le rapport.

IV RENDU DU PROJET

- Le projet est à réalisé **individuellement.**
- Le rendu est sous la forme **PRENOM_NOM.zip**, archive contenant les fichiers **.h, .c et rapport.pdf**
- Les projets sont à déposer sur l'espace de rendu de *Campus Efrei*, ceux rendus uniquement par mail et/ou en retard se verront attribuer un **malus de 1 point par jour de retard.**
- Un rapport doit fournir votre analyse et votre conception.
L'organisation du travail et la chronologie des développements est à décrire.
- **Une bonne idée est de réaliser le rapport en même temps que le développement.**

Tout plagiat se verra sanctionné par un ZERO et sera l'objet d'un rapport à la direction des études.

ANNEXE METHODE DE HORNER

On veut calculer $P(x)$, où x est un réel (long double)
et P est un polynôme de degré n , représenté par un tableau :

si $P(X) = a_0 + a_1X + \dots + a_nX^n$

alors pour tout i dans $[0, n]$ on a $P[i] = a_i$.

```
long double Horner(long double p[], long n, long double x) {  
    long double r = p[n];  
    for(long i = n - 1; i >= 0; i--) {  
        r = (r * x) + p[i];  
    }  
    return r;  
}
```

La complexité de l'algorithme de Horner est en $O(n)$.