Techniques de programmation – TP2

Introduction

Représentation mathématique

Un polynôme p_n de degré n à coefficients a_i, $i = 0..n$ dans ℝ est une expression algébrique de la forme :

$$p_n(x) = \sum_{i=0}^{n} a_i x^i \,, a_i \in \mathbb{R}$$

Dans cette expression :

- n est le degrè de p_n
- a_i, $i = 0..n$: constantes réelles appelées coefficients du polynômes.
- a_0,: est appelé coefficient indépendant,
- x : est une variable réelle appelée variable du polynôme

Exemples de polynômes

- P1(x)=1 - 3x
- P2(x)=2 + 5x^1000

Intérêt des polynômes

Les polynômes permettent de représenter un grand nombre de courbes ce qui fait qu'ils sont largement utilisés dans plusieurs domaines :

- En animation vidéo pour représenter les contours des objets virtuels,
- En astraunotique et en balistique, pour représenter les trajectoires des sondes et des projectiles,
- En informatique, dans le calcul des codes de controles de transmission,
- En finance, pour modéliser les comportements des marchés, etc.

Partie 1 – Manipulations de base sur les polynômes

Sur l'ensemble des polynômes, on souhaite définir des opérations suivantes :

- ✓ La Saisie des données d'un polynôme,
- ✓ ✓ Afficher un polynôme sous la forme suivante :

  P1(x)= 1 * x ^ 0 - 3 * x ^ 1

  P2(x)= 2 * x ^ 0 + 5 * x ^ 1000
- ✓ Evaluer la valeur du polynôme étant donnée la valeur de la variable x,
- ✓ Faire l'addition de deux polynômes P et Q fournis en paramètres,
- ✓ Changer le signe d'un polynôme en changeant le signe de tous ses coefficients,

- ✓ Faire la soustraction de deux polynômes,
- ✓ Faire la comparaison de deux polynômes. On ne peut que dire si deux polynômes sont égaux ou non.
- ✓ Calculer la dérivée d'un polynôme,
- ✓ Tracer la courbe représentative du polynôme dans un intervalle I=[a b] en utilisant le grapheur Gnuplot

Travail à faire

1. Analyser ce cas et proposer une structure de données pour représenter un polynôme.
2. Donner une abstraction des différentes opérations en se basant sur la structure choisie.
3. Réaliser les différentes opérations
4. Donner une écriture de test qui définit deux polynômes p1 et p2 de degrés inférieurs à 10 et qui les trace dans le même graphique en plus de leur somme p1+p2 dans l'intervalle [0 10].

Partie 2 : Calcul numérique d'une racine d'un polynôme

On souhaite implémenter une solution qui calcule une racine approché d'un polynôme en partant d'une 1ère valeur choisie aléatoirement. On sait qu'il n'existe pas de méthode de calcul des racines de polynômes de degré quelconque. Il existe en revanche des méthodes de calcul numérique permettant d'approcher les valeurs des racines. La méthode de Newton-Raphson, proposée par I. Newton et J. Raphson, puis généralisée par T. Simpson, en est un exemple. Cette méthode consiste à prendre une première estimation (aléatoirement ou non) d'une valeur de x que l'on espère proche d'une racine, puis à raffiner cette valeur en suivant la pente de la fonction donnée par sa dérivée. Formellement, il s'agit de calculer la suite par la forme itérative suivante :

$$\begin{cases} x_0 = r \\ x_{k+1} = x_k - \frac{p(x_k)}{p'(x_k)} \end{cases}$$

avec la formule donnant la dérivée p' :

$$p'(x) = \sum_{j=0}^{n-1} (j+1).a_{j+1}.x^j$$

Où r est l'estimation initiale de la racine et x_k l'estimation raffinée de cette racine. Plus la valeur de k est élevée, plus l'approximation sera précise (sous réserve que la méthode converge, c'est-à-dire trouve une solution).
La Figure 1 montre un exemple de calcul de racine par la méthode de Newton-Raphson.

```
Polynome : p(x) = 1.00 + 1.00.X^1 + 2.00.X^2 -2.00.X^3
Derivee : p'(x) = 1.00 + 4.00.X^1 -6.00.X^2

k = 0     x0 = 1.0000000000     p(x0) = 2.0000000000
k = 1     x1 = 3.0000000000     p(x1) = -32.0000000000
k = 2     x2 = 2.2195122242     p(x2) = -8.7956943512
k = 3     x3 = 1.7725619078     p(x3) = -2.0821790695
k = 4     x4 = 1.5790797472     p(x4) = -0.3087827861
k = 5     x5 = 1.5386877060     p(x5) = -0.0120629631
k = 6     x6 = 1.5369768143     p(x6) = -0.0000213105
k = 7     x7 = 1.5369738340     p(x7) = -0.0000005270
k = 8     x8 = 1.5369737148     p(x8) = 0.0000002089
k = 9     x9 = 1.5369737148     p(x9) = 0.0000002089
k = 10    x10 = 1.5369737148    p(x10) = 0.0000002089
```

Figure 1 : Calcul d'une racine par la méthode de Newton-Raphson

Dans cette partie, on cherche à réaliser un algorithme pour calculer les racines de polynômes de degré n à l'aide de la méthode de Newton-Raphson décrite ci-dessus.

Travail à faire

5. Analyser ce cas et proposer une abstraction des opérations à prévoir pour calculer une racine approchée.
6. Réaliser les différentes opérations et donner une écriture de test