

R2.01 – Développement orienté objets**TD n° 5 – Classe *Polynôme* (partie 1/3)**

Un polynôme $P(x)$, en mathématiques est une somme de monômes $a_i x^i$ où x représente une variable, a_i le coefficient du monôme et i son exposant.

On le note : $P(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$

On en propose la spécification fonctionnelle suivante :

Type Polynôme**Opérations**

<i>polynômeNul</i> :	\rightarrow Polynôme
<i>setMonôme</i> :	Polynôme x Monôme \rightarrow Polynôme
<i>getMonôme</i> :	Polynôme x Entier \rightarrow Monôme
<i>somme</i> :	Polynôme x Polynôme \rightarrow Polynôme
<i>produit</i> :	Polynôme x Monôme \rightarrow Polynôme
<i>dérivée</i> :	Polynôme \rightarrow Polynôme

Préconditions

getMonôme(p, e) ssi $e \geq 0$

Propriétés (des trois premières opérations)

(P1) *getMonôme*(*polynômeNul*, e) = *unMonôme*(0, e)

(P2) *getMonôme*(*setMonôme*(p, *unMonôme*(c, e)), e') =
 si $e=e'$ alors *unMonôme*(c, e)
 sinon *getMonôme*(p, e')

Propriétés des autres opérations de *Polynôme* :

Si $P(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$

et $Q(x) = b_0 + b_1 x + b_2 x^2 + \dots + b_n x^n$

- $P(x) + Q(x) = (a_0+b_0) + (a_1+b_1) x + (a_2+b_2) x^2 + \dots + (a_n+b_n) x^n$
- $P'(x) = (a_0)' + (a_1 x)' + (a_2 x^2)' + \dots + (a_n x^n)'$
- Pour un monôme m :
 $P(x) * m = (a_0 m) + (a_1 x m) + (a_2 x^2 m) + \dots + (a_n x^n m)$

Remarque :

Par convention, le degré du polynôme nul est égal à $-\infty$. En pratique, on conviendra qu'un polynôme nul est tel que tous ses coefficients sont nuls.

Nous souhaitons construire et utiliser la classe Java *Polynôme* mettant en œuvre ce TAD.

Remarque : En annexe figure un extrait de la documentation au format javadoc de la classe *Monôme* et de la classe *Polynôme* que nous souhaitons construire.

Questions :

1) Représentation d'un polynôme

Pour incarner un polynôme dans une classe Java, nous avons choisi de stocker les coefficients de ses monômes dans un tableau *coefficients* où *coefficients[i]* représente le coefficient du monôme d'exposant *i* du polynôme. Donner sa définition en Java.

2) Fournir le corps du constructeur puis des méthodes *setMonôme()* et *getMonôme()* de la classe *Polynôme*.

3) Algorithmes des autres méthodes de la classe *Polynôme*

Proposer un algorithme pour l'écriture des méthodes : *dérivée()*, *produit()* et *somme()* de la classe *Polynôme* en s'appuyant sur les propriétés des opérations.

4) Application : les polynômes d'Hermite¹

Les *polynômes d'Hermite sous forme probabiliste* peuvent s'obtenir par la formule de récurrence suivante :

$$\begin{aligned}H_0(x) &= 1 \\H_1(x) &= x \\H_n(x) &= x H_{n-1}(x) - (n-1)H_{n-2}(x) \text{ pour } n > 1\end{aligned}$$

Ecrire une application Java qui construit et affiche les éléments d'un tableau contenant les 10 premiers polynômes d'Hermite en utilisant la formule de récurrence ci-dessus.

Les 10 premiers polynômes à obtenir sont :

$$\begin{aligned}H_0(x) &= 1 \\H_1(x) &= x \\H_2(x) &= x^2 - 1 \\H_3(x) &= x^3 - 3x \\H_4(x) &= x^4 - 6x^2 + 3 \\H_5(x) &= x^5 - 10x^3 + 15x \\H_6(x) &= x^6 - 15x^4 + 45x^2 - 15 \\H_7(x) &= x^7 - 21x^5 + 105x^3 - 105x \\H_8(x) &= x^8 - 28x^6 + 210x^4 - 420x^2 + 105 \\H_9(x) &= x^9 - 36x^7 + 378x^5 - 1260x^3 + 945x\end{aligned}$$

¹ Les *polynômes d'Hermite* sont des polynômes orthogonaux servant dans le calcul vectoriel et à des problèmes d'interpolation

On peut remarquer que dans la formule de récurrence :

- le facteur x associé au polynôme $H_{n-1}(x)$ correspond au monôme x^1
- le facteur $(n - 1)$ associé au polynôme $H_{n-2}(x)$ correspond au monôme $(n - 1)x^0$

ANNEXE

Class Monôme

java.lang.Object

Monôme

```
public class Monôme
extends java.lang.Object
```

Constructor Summary

Constructors

Constructor and Description

[Monôme](#)(float coefficient, int exposant)
construit un monôme

Method Summary

All Methods

Modifier and Type	Method and Description
<u>Monôme</u>	<u>dérivée</u> () calcule la dérivée d'un monôme
boolean	<u>estNul</u> () teste si un monôme est nul
float	<u>getCoefficient</u> () retourne le coefficient d'un monôme
int	<u>getExposant</u> () retourne l'exposant d'un monôme
<u>Monôme</u>	<u>produit</u> (<u>Monôme</u> m) calcule le produit de deux monômes
<u>Monôme</u>	<u>somme</u> (<u>Monôme</u> m) calcule la somme de deux monômes
java.lang.String	<u>toString</u> () produit une version unicode d'un monôme

Class Polynôme

java.lang.Object

Polynôme

```
public class Polynôme
extends java.lang.Object
```

Constructor Summary

Constructors

Constructor and Description

[Polynôme](#) ()
crée un polynôme nul

Method Summary

All Methods

Modifier and Type	Method and Description
<u>Polynôme</u>	<u>dérivée</u> () calcule la dérivée d'un polynôme
<u>Monôme</u>	<u>getMonôme</u> (int exposant) accède à un monôme du polynôme
<u>Polynôme</u>	<u>produit</u> (<u>Monôme</u> m) calcule le produit d'un polynôme et d'un monôme
void	<u>setMonôme</u> (<u>Monôme</u> m) positionne un monôme dans un polynôme
<u>Polynôme</u>	<u>somme</u> (<u>Polynôme</u> p) calcule la somme de deux polynômes
java.lang.String	<u>toString</u> () produit une version unicode d'un polynôme