POLYNOMES : METHODE DE HORNER

Un polynôme $P = a_n.x^n + a_{n-1}.x^{n-1} + \ldots + a_2.x^2 + a_1.x + a_0$ est déterminé par la liste $(a_0, a_1, \ldots a_{n-1}, a_n)$ de ses coefficients.

1) Saisir le degré n d'un polynôme P, ses coefficients, et l'afficher sous la forme :
   $P = a_n.x^n + a_{n-1}.x^{n-1} + \ldots + a_2.x^2 + a_1.x + a_0$
2) Saisir une valeur de x et calculer la valeur du polynôme P en x, valeur que l'on note $P(x)$.
   Afin d'améliorer ce calcul, utiliser la méthode de Hörner, basée sur l'égalité suivante (il s'agit d'une réécriture) :
   $a_n.x^n + a_{n-1}.x^{n-1} + \ldots + a_2.x^2 + a_1.x + a_0 = a_0 + x\ (\ a_1 + x\ (\ldots(\ a_{n-2} + x\ (\ a_{n-1} + x\ a_n\ ))\ldots))$
   Quelle relation existe-t-il entre le degré n du polynôme stocké et la taille utile du tableau?
3) Soit 2 polynômes P et Q, dont les coefficients sont déjà saisis et dont les degrés respectifs sont donnés par des variables p et q. Selon les valeurs des coefficients, quel sera le degré de $P+Q$?
4) Calculer la somme des polynômes $P(x) + Q(x)$, ainsi que le degré de ce polynôme.
5) Calculer le polynôme dérivé P' d'un polynôme P.
6) Calculer le polynôme intégral Π d'un polynôme P (c'est à dire le polynôme Π tel que $\Pi' = P$), sachant que $\Pi(0) = K$, K étant une valeur réelle arbitraire.

<u>Indication:</u> le type Polynome est défini et les fonctions sont déclarées de la manière suivante:

typedef struct polynome	void saisirPolynome(Polynome * P);
{	void afficherPolynome(Polynome P);
long double * a; // tableau des coefficients	long double Horner(Polynome P, long double x);
long n; // degre	void sommePolynome(Polynome P, Polynome Q, Polynome * S);
} Polynome;	void derivePolynome(Polynome P, Polynome * Pprim);
	void integralPolynome(Polynome P, Polynome * Pintegral);

L'algorithme de la méthode Horner est implanté de la manière suivante pour un polynôme représenté par un tableau pour ses coefficients et un entier pour son degré:

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

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

La sortie attendue est la suivante:

```
SAISIE DU POLYNOME P
degre : 5
coefficient a0 : 3
coefficient a1 : 2
coefficient a2 : 1
coefficient a3 : 3
coefficient a4 : 4
coefficient a5 : 3
POLYNOME P DE DEGRE 5 : (3.00, 2.00, 1.00, 3.00, 4.00, 3.00)

Methode de Horner : calcul de P(X) pour X : 7
P(7.00) = 61120.00

SAISIE DU POLYNOME Q
degre : 8
coefficient a0 : 8
coefficient a1 : 8
coefficient a2 : 8
coefficient a3 : 3
coefficient a4 : 3
coefficient a5 : 3
coefficient a6 : 3
coefficient a7 : 3
coefficient a8 : 32
POLYNOME Q DE DEGRE 8 : (8.00, 8.00, 8.00, 3.00, 3.00, 3.00, 3.00, 3.00, 32.00)

CALCUL DU POLYNOME S = P+Q
POLYNOME P DE DEGRE 5 : (3.00, 2.00, 1.00, 3.00, 4.00, 3.00)
POLYNOME Q DE DEGRE 8 : (8.00, 8.00, 8.00, 3.00, 3.00, 3.00, 3.00, 3.00, 32.00)
POLYNOME S = P+Q DE DEGRE 8 : (11.00, 10.00, 9.00, 6.00, 7.00, 6.00, 3.00, 3.00, 32.00)

CALCUL DU POLYNOME DERIVE P' DU POLYNOME P
POLYNOME P DE DEGRE 5 : (3.00, 2.00, 1.00, 3.00, 4.00, 3.00)
POLYNOME P' DE DEGRE 4 : (2.00, 2.00, 9.00, 16.00, 15.00)

CALCUL DU POLYNOME INTEGRAL DU POLYNOME P
POLYNOME P DE DEGRE 5 : (3.00, 2.00, 1.00, 3.00, 4.00, 3.00)
POLYNOME INTEGRAL DE DEGRE 6 : (0.00, 3.00, 1.00, 0.33, 0.75, 0.80, 0.50)
```