TD 5
Diviser pour régner - Polynômes

On considère des polynômes sur le corps $\mathbb{R}$ des nombres réels. On représentera un polynôme P par un tableau p, de la façon suivante : $P(X) = \sum_{i=0}^{n-1} p[i] X^i$; la taille du tableau (et du polynôme) est n, en revanche le degré du polynôme dépend de la nullité éventuelle des coefficients.

On supposera que les opérations élémentaires sur $\mathbb{R}$ (i.e. addition, multiplication, inversion) prennent un temps constant. On exprimera la complexité des algorithmes en fonction de la taille n des polynômes. Par défaut il s'agira de la complexité en temps dans le pire des cas, et on se contentera de la majoration par O.

1 Complexité d'algorithmes de type *Diviser pour régner*

Exercice 1. On s'intéresse à une fonction T de complexité en temps qui vérifie $T(2n) \leq aT(n)+bn$ avec $a > 2$. Bien sûr, on supposera T croissante. Le but de cette partie est de majorer T par un O d'une fonction élémentaire.

1. Montrer le lemme suivant : si f et g sont croissantes, $g(2^n) \in O(f(2^n))$ et $f(2n) \in O(f(n))$, alors $g(n) \in O(f(n))$.
2. On pose $g(n) = T(n) + \frac{b}{a-2}n$. Montrer que $g(2n) \leq ag(n)$. En déduire que $g(2^n) \leq a^n g(1)$.
3. On pose $f(n) = n^{\log_2 a}$. Montrer que $f(2n) \in O(f(n))$.
4. Conclure.

2 Algorithmes élémentaires sur les polynômes

Exercice 2. Degré. Normalisation.

1. Rappeler la définition du degré d'un polynôme.
2. Donner un algorithme `degré`(P) qui renvoie le degré du polynôme P. Quelle est sa complexité ?
3. Donner un algorithme `normalisation`(P) qui renvoie le polynôme unitaire proportionnel à P. Quelle est sa complexité ?

Exercice 3. Dérivée.

1. Rappeler la formule qui donne P'.
2. Donner un algorithme `dérivée`(P) qui renvoie P'. Quelle est sa complexité ?

Exercice 4. Évaluation.

1. Proposer un algorithme `évaluation`(P, x) qui renvoie $P(x)$. Quelle est sa complexité exacte ?
2. Même question en utilisant le schéma de Horner :

$$P(x) \quad = \quad (\dots(p[n-1]\,x + p[n-2])\,x + \dots)\,x + p[0] \quad .$$

Exercice 5. Somme.

1. Effectuer à la main la somme suivante : $\left(3X^4 + X^2 + 1\right) + \left(2X^3 + 5X^2 + 7X\right)$.
2. On suppose que P et Q sont de même taille n. Donner un algorithme `somme` (P, Q) qui renvoie $P + Q$. Quelle est sa complexité ?

Exercice 6. Produit.

1. Effectuer à la main le produit suivant : $\left(3X^4 + X + 7\right) \cdot \left(4X^2 + 3\right)$.
2. On suppose que P et Q sont de même taille n. Donner un algorithme naïf `produit` (P, Q) qui renvoie PQ. Quelle est sa complexité ?
3. On va tenter d'améliorer cette complexité en utilisant le paradigme *Diviser pour régner*. Pour ce faire, on va découper en 2 les polynômes P et Q. Soit Soit $m = \lceil \frac{n}{2} \rceil$; on fait le découpage suivant :

$$\begin{aligned} P &= A_1 X^m + B_1 \\ Q &= A_2 X^m + B_2 \end{aligned}$$

avec le degré de A_1, B_1, A_2, B_2 inférieur ou égal à $\lceil \frac{n}{2} \rceil - 1$ (et donc de taille inférieure ou égale à $\lceil \frac{n}{2} \rceil$).

   (a) En déduire un algorithme `produit_dpr` (P, Q) de type *Diviser pour régner* qui renvoie PQ. Prouver sa terminaison.

   (b) Quelle est sa complexité ? Indication : exprimer $T(n)$ en fonction de $T\left(\lceil \frac{n}{2} \rceil\right)$ et utiliser le résultat de la partie 1.

   (c) On peut obtenir mieux en faisant une multiplication auxiliaire de moins. Comment cela ? Quelle est alors la complexité du produit ?

   (d) Note de la rédaction : on peut faire un produit de polynômes en temps $O(n \ln n)$ grâce à la transformée de Fourier rapide, mais ça commence à devenir un peu plus compliqué...

Exercice 7. Division euclidienne.

1. Compléter l'énoncé suivant : "Soit $A, B \in \mathbb{R}[X]$. Si $B \neq 0$, alors il existe un et un seul couple (Q, R) de polynômes de $\mathbb{R}[X]$ tels que..."
2. Effectuer à la main la division euclidienne de $X^4 + X^3 + 5X^2 + 5X + 2$ par $X^2 + 2$.
3. Donner un algorithme `diveuclid` (A, B) qui renvoie le couple (Q, R) mentionné au point 1. Prouver sa terminaison. Quelle est sa complexité en temps ? en espace ?

Exercice 8. PGCD.

1. Rappeler le principe de l'algorithme d'Euclide pour calculer le PGCD.
2. Faites-le tourner pour calculer le PGCD de $2X^3 + X^2 + 2X + 1$ et de $2X^3 + 3X^2 + 3X + 1$.
3. Écrire l'algorithme d'Euclide `pgcd` (A, B) qui renvoie le PGCD de A et B (supposés non nuls). Prouver sa terminaison. Quelle est sa complexité en temps ? en espace ?
4. En déduire un algorithme `bezout` (A, B) qui renvoie 2 polynômes U et V tels que $AU + BV =$ PGCD (A, B). Prouver sa terminaison. Quelle est sa complexité en temps ? en espace ?