Lycée Saint-Louis — TP
MPSI

Manipulation de polynômes en Caml

Dans toute la suite, on représente le polynôme réel $a_nX^n + \ldots + a_1X + a_0$ par la liste $[a_0; \ldots; a_n]$. On dit que cette représentation est normale si $a_n \neq 0$.
Ainsi, $X^2 - 2$ a pour représentation normale $[-2.\,;\,0.\,;\,1.]$, mais peut aussi être représenté par $[-2.\,;\,0.\,;\,1.\,;\,0.\,;\,0.]$. Le polynôme nul a pour représentation normale $[\,]$.
Sauf mention du contraire, les polynômes arguments ou renvoyés ne seront pas forcément en représentation normale.

1 Premières opérations sur les polynômes

1. Écrire une fonction `est_normal` prenant en argument un polynôme et renvoyant `true` si celui-ci est en représentation normale et `false` sinon.
2. Écrire une fonction `dilatation` prenant en argument un polynôme p et un flottant a et renvoyant ap.
3. Écrire une fonction `somme` prenant en argument deux polynômes et renvoyant leur somme.
4. Écrire une fonction `evaluation` prenant en argument un polynôme p et un flottant v et renvoyant l'évaluation de p en v.
5. Écrire une fonction `produit` prenant en argument deux polynômes et renvoyant leur produit.
6. Déterminer la complexité des fonctions précédentes.

2 Algorithme de Karatsuba

Soient A et B deux polynômes de $\mathbb{R}_{2^m-1}[X]$ que l'on considèrera représentés par des listes de longueur $n = 2^m$. On veut calculer le produit AB de manière plus efficace que par le programme de la partie précédente.

Pour utiliser une stratégie de type *diviser pour régner*, on veut se ramener aux produits de polynômes de taille deux fois plus petite.
Pour cela, on utilise la division euclidienne par $X^{n/2} = X^{2^{m-1}}$ des deux polynômes :

$$A = A_1X^{n/2} + A_0 \qquad B = B_1X^{n/2} + B_0$$

et on utilise le fait que :

$$AB = A_1B_1X^n + (A_1B_0 + A_0B_1)X^{n/2} + A_0B_0 \qquad (1)$$

1. Écrire une fonction `foisXn` prenant en argument un polynôme p et un entier n et calculant le produit de p et X^n.
2. (a) Déterminer la relation de récurrence que vérifierait la complexité d'un algorithme récursif utilisant directement l'égalité (1), et faisant donc 4 appels récursifs.
   (b) En déduire la complexité asymptotique d'un tel algorithme. Est-elle meilleure que la complexité obtenue dans la partie 1 ?
3. Expliquer comment l'on peut descendre le nombre d'appels récursifs à 3 en utilisant le produit $(A_0 + A_1)(B_0 + B_1)$.
4. Écrire une fonction `karatsuba` implémentant cet algorithme.
5. Déterminer la relation de récurrence vérifiée par la complexité de cette fonction, et en déduire la complexité de cet algorithme.
6. Lorsque la longueur de la liste représentant un polynôme n'est pas une puissance de 2, on la complète avec des zéros pour obtenir une longueur puissance de 2 pour pouvoir appliquer la fonction `karatsuba`. Écrire une fonction final `karatsuba` ne nécessitant pas que les polynômes utilisés soient représentées par deux listes de même longueur et dont la longueur soit une puissance de 2. La complexité est-elle modifiée ?
7. Écrire une fonction `expo_rapide` prenant en argument un entier m et un polynôme p et calculant p^m en utilisant l'algorithme d'exponentiation rapide.
8. Déterminer la relation de récurrence vérifiée par la complexité de cette fonction et montrer qu'elle est en $O((nm)^\alpha)$, où $n = \text{len}(p)$ et $\alpha = \log_2(3)$.

3 Autres opérations sur les polynômes

1. (a) Écrire une fonction `normalisation` prenant un polynôme en argument et renvoyant sa représentation normale.
   (b) Démontrer que cette fonction termine et est correcte.
   (c) Déterminer sa complexité.
2. (a) Écrire une fonction `derivation` prenant en argument un polynôme et renvoyant son polynôme dérivé.
   (b) Démontrer la terminaison et la correction de cette fonction.
   (c) Déterminer sa complexité.