Travaux pratiques

Décomposition en sous-problèmes

1. On représente un polynôme par un tableau de ses coefficients $[|a_0; ....; a_n|]$.
   (a) Quel est le coût de l'accès au coefficient ? de l'addition de deux polynômes ? de la multiplication naïve de deux polynômes ? *On admet qu'une multiplication est bien plus coûteuse qu'une addition, et on ne considère désormais que ces premières.*
   (b) Pour multiplier deux polynômes P et Q de degré n, on les écrit sous la forme $P = P_1 X^{\lceil \frac{n}{2} \rceil} + P_2$ et $Q = Q_1 X^{\lceil \frac{n}{2} \rceil} + Q_2$, puis on observe que, en notant $m = \lceil \frac{n}{2} \rceil$, $PQ = P_1 Q_1 X^{2m} + (P_1 Q_2 + P_2 Q_1) X^m + P_2 Q_2$. Quelle est la complexité d'un algorithme utilisant cette relation ?
   (c) Observons maintenant que $PQ = P_1 Q_1 X^{2m} + ((P_1 + P_2)(Q_1 + Q_2) - P_1 Q_1 - P_2 Q_2) X^m + P_2 Q_2$. Combien d'appels récursifs à la fonction de multiplication seront nécessaires en utilisant cette relation ? Quelle est la complexité de la fonction qui l'utilise ?
   C'est l'algorithme de Karatsuba, historiquement le premier algorithme de multiplication rapide. Implémenter cet algorithme.
   (d) Proposer un algorithme analogue pour la multiplication de grand entiers entre eux.

2. On souhaite multiplier entre elles deux matrices carrées.
   (a) Quel est le coût de l'addition de telles matrices ? De la multiplication ?
   (b) Posons $M = \begin{pmatrix} A_1 & B_1 \\ C_1 & D_1 \end{pmatrix}$ et $N = \begin{pmatrix} A_2 & B_2 \\ C_2 & D_2 \end{pmatrix}$. L'observation suivante permet-il d'accélérer le produit matriciel ?
   $MN = \begin{pmatrix} A_1 A_2 + B_1 C_2 & A_1 B_2 + B_1 D_2 \\ C_1 A_2 + D_1 C_2 & C_1 B_2 + D_1 D_2 \end{pmatrix}$
   (c) L'algorithme de **Strassen** réécrit $MN = \begin{pmatrix} X & Y \\ Z & T \end{pmatrix}$ où

$$\begin{array}{lll} M_1 = (B_1 - D_1)(C_2 + D_2) & M_5 = A_1(B_2 - D_2) & X = M_1 + M_2 - M_4 + M_6 \\ M_2 = (A_1 + D_1)(A_2 + D_2) & M_6 = D_1(C_2 - A_2) & Y = M_4 + M_5 \\ M_3 = (A_1 - C_1)(A_2 + B_2) & M_7 = (C_1 + D_1)A_2 & Z = M_6 + M_7 \\ M_4 = (A_1 + B_1)D_2 & & T = M_2 - M_3 + M_5 - M_7 \end{array}$$

   Quelle est la complexité de cet algorithme ? L'implémenter.