Université de Rouen Normandie UFR Sciences et Techniques

Rapport du projet Calcul Symbolique

Projet réalisé par :

FOFANA Abdoulaye

BEN AHMED Mohamed Dhia

2021 – 2022

Table des matières

[1- Polynômes à coefficient dans F2 2](#_Toc99897778)

[1. Le produit par la méthode Karatsuba 2](#_Toc99897779)

[2. La division rapide par méthode de Newton 2](#_Toc99897780)

[3. Algorithme d’Euclide 2](#_Toc99897781)

[4. Spécification des fonctions : 2](#_Toc99897782)

[2- Registre à décalage à rétroaction linéaire 2](#_Toc99897783)

# Polynômes à coefficient dans F2

## Le produit par la méthode Karatsuba

L'algorithme de Karatsuba pour la multiplication de deux polynômes a été introduit en 1962. Il permet d'économiser les multiplications de coefficients au prix d'additions supplémentaires par rapport à la méthode de multiplication ordinaire.  
La méthode de Karatsuba de base s'effectue comme suit. Considérons deux polynômes de degré 1 A(x) et B(x) avec n = 2 coefficients.

A(x) = 𝑎0 + 𝑎1X  
 B(x) = 𝑏0+ 𝑏1X

Soit 𝐷0, 𝐷1, 𝐷0,1 des variables auxiliaires avec :

𝐷0 = 𝑎0 𝑏0  
 𝐷1 = 𝑎1 𝑏1  
 𝐷0,1 = (𝑎0 +𝑎1 )( 𝑏0 + 𝑏1)

Ensuite, le polynôme C(x) = A(x) B(x) peut être calculé de la manière suivante :

C(x) = 𝐷1 𝑋2 +( 𝐷0,1 - 𝐷0 -𝐷1)X + 𝐷0

Cette méthode nécessite trois multiplications et quatre additions. La méthode classique de multiplication nécessite 𝑛2 multiplications et (𝑛 − 1)2 additions, c'est-à-dire quatre multiplications et une addition. Il est clair que la méthode de Karatsuba peut également être utilisée pour multiplier des nombres entiers.  
La méthode de Karatsuba peut être généralisée pour des polynômes de degré n

## La division rapide par méthode de Newton

|  |
| --- |
| * entrées : les polynômes A (de degré n) et B (de degré m et de coefficient dominant bm) * sortie : le couple (Q, R) tel que A = BQ + R et deg R < m 1) si n < m renvoyer (0, A) 2) P ← newton( 1/bm( Revm(B)), n − m + 1) 3) Q ← 1/bm( Revn−m)(Revn(A)P [Xn−m+1]) 4) renvoyer (Q, A − BQ) |

## Algorithme d’Euclide

|  |
| --- |
| * entrée : e, f ∈ N * en sortie : pgcd(e, f )   début 1. si f = 0 alors retourner e 2. sinon retourner pgcd(f , e mod f ) fin |

## Spécification des fonctions :

**add\_poly :** la somme de deux polynôme

**mult\_coeff** : permettant de multiplier les coefficients de deux polynômes

**degre** : permettant de retourner le degré d'un polynôme

**multXn** : permettant de multiplier un monôme X^n par un polynôme

**cut** : retourne le couple cut1 et cut2

**cut1** : retourne la moitié du polynôme p dont le degré est inférieur à n

**cut2** : retourne la moitié du polynôme p dont le degré est supérieur à n

**multinaive** : Multiplication de deux polynômes par la méthode de la multiplication naïve

**karatsuba** : Multiplication de deux polynômes par la méthode de Karatsuba

**moduloXn** : Division de deux polynômes par la méthode de Newton

**quotient** : retourne le quotient

**reste** : retourne le reste

# Registre à décalage à rétroaction linéaire

Un registre à décalage à rétroaction linéaire (LFSR : Linear Feedback Shift Register) est un dispositif permettant de produire une suite de bits ultimement périodique à coefficient dans F2. C’est un dispositif relativement simple et léger qui peut être réalisé électroniquement. Les LFSR sont utilisés en cryptographie pour engendrer des suites de nombres pseudo-aléatoires. Ils sont définis par une suite récurrente :  
  
 r0, . . . , r`−1 ∈ F2  
 rn = α1 ⊗ rn−1 ⊕ · · · ⊕ α` ⊗ rn−l si n ≥ l

Où l > 0 est la longueur du LFSR. Les valeurs αi ∈ F2 sont appelés branchements. On note Vi (i ≥ 0) le vecteur (ou encore registre) qui indique l’état du LFSR :

Vi = (ri, ri+1 , ... ,ri+l−1)

Le registre V0 correspond à l’état initial du LFSR.

Problèmes rencontrés

Lors de la construction des algorithmes de la première partie ayant déjà vu ces derniers en TP nous avons donc pas eu beaucoup de soucis (choix de la structure adapté pour la représentation du corps F2), or malheureusement quand on a essayé de construire les algorithmes de la deuxième partie, et malgré la bonne compréhension du principe des LFSR et de la méthode dont ils fonctionnent on a trouvé pas mal de problèmes c’est pourquoi on ne les a pas implémenté.