Projet de Calcul Symbolique
L3 Informatique
2021-2022

Projet

Projet LFSR

Dombry Baptiste
Dourlen Maxime
Groupe TP3

Sommaire :

I. Polynômes a coefficient dans F2
II. Registre a décalage à rétroaction linéaire
III. Application a la cryptographie

I. Polynômes à coefficient dans F2

Exercice 1
1)

F2 est un corps commutatif si

$$\forall (a,b,c) \in K^3 \quad a \times (b+c) = a \times b + a \times c \quad \text{et} \quad (b+c) \times a = b \times a + c \times a$$

On crée deux tables de vérité pour vérifier.

a	b	c	a * (b + c)
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

a	b	c	a * b + a * c
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

On voit que les deux tables sont identiques pour tous les éléments de F2. On en conclut que dans F2 la multiplication est distributive sur l’addition donc F2 est un corps commutatif.

2)

Les opérations ⊕ et ⊗ sont assimilables à des opérations logiques :
- Le ⊕ a un OU exclusif (xor)
- Le ⊗ a un ET (and)

Exercice 2

1)

La structure de donnée que l’on a choisi pour l’implantation des polynômes dans F2 est une liste d’entiers.

3)

La fonction ordre permet de trouver l’ordre d’un polynôme, pour cela nous allons d’abord à un ordre du degré du polynôme ensuite on effectue la soustraction, si le résultat est la constante alors nous renvoyons l’ordre sinon nous augmentons l’ordre jusqu’à ce que le degré du résultat précédent soit supérieur ou égale au degré du polynôme.

4)

La fonction « irreductible » permet de savoir si un polynôme est irréductible. Pour cela, nous générons tous les polynômes d’un certain degré grâce à la fonction « generate_degre_poly » qui renvoie la liste de tous les polynômes du degré placé en paramètre. Ensuite on divise le polynôme p par tous ses diviseurs possibles et si on trouve un reste égale à une constante on envoie false si on ne trouve pas alors on envoie true.

5)

La fonction « create_primitf » permet de créer un polynôme primitif, ce polynôme respectera donc les deux conditions, il sera irréductible et son ordre sera de 2^n-1. Pour cela, nous générons la liste de tous les polynômes de degré n, si un polynôme dans cette liste respecte les conditions alors le polynôme est renvoyé sinon la fonction génère la liste n+1 et cherche le polynôme qui respectera les conditions.

II. Registre à décalage à rétroaction linéaire

Exercice 3

1)

La structure de donnée que l'on a choisi pour l'implantation des LFSR est une liste de couple (entier, booléen). Les entiers représentent les R_i et les booléens représentent les Alpha_i.

2)

La fonction qui fait le calcul de la nième valeur rn de la suite ri est la fonction
get_rn_from_ri l i n
avec l LFSR
i et n entiers.

3)

On montre par récurrence que $M^n V_0 = V_n$

Base : $n = 0$

On sait que M^0 = matrice identité

Si $M = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ \alpha_3 & \alpha_2 & \alpha_1 \end{pmatrix}$, $M^0 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Or $M^0 V_0 = V_0$ car la matrice identité multipliée par un vecteur est égale à ce vecteur.

Hérédité :

On suppose la propriété vraie au rang n.
Démontrons que la propriété est vraie au rang $n+1$
c'est-à-dire que $M^{n+1} V_0 = V_{n+1}$

$M^n V_0 = V_n$
$\Leftrightarrow (M^n V_0) \times M = V_n \times M$
$\Leftrightarrow M^{n+1} V_0 = V_n \times M$

Or, on sait que $V_n \times M$ est égale à V_{n+1}

Donc, on a : $M^{n+1} V_0 = V_{n+1}$

Conclusion :

La propriété est vraie au rang 0 et elle est héréditaire donc
$\forall n \geq 0$, $M^n V_0 = V_n$

4)

Soit Vi = (ri, ri+1, ri+2 ... , ri+l-1) de ieme registre.
Celui-ci détermine complètement les registres ultérieurs.
Ce registre peut prendre au plus 2^L états

S’il atteint l’état 0 = (0,..., 0) alors les registres successifs sont tous nuls et la
suite elle-même est nulle à partir de là.
S’il n’est jamais nul, parmi [V0,V1,...,V(2)^L-1], au moins deux registres sont identiques.
Supposons Vi0= Vi0+T ; alors la suite des registres [V_i0,V_i0+1,...,V_i0+T −1] se répète indéfiniment.
On a donc si = si+T pour tout i ≥ i0 avec T ≤ 2^L −1.

5)

Pour le premier LFSR on a :
V0 = [1;0;0;1;0;0;1;0;0;1]
V1 = [0;1;0;0;1;0;0;1;0;0]
V2 = [0;0;1;0;0;1;0;0;1;0]
V3 = [1;0;0;1;0;0;1;0;0;1]
On remarque que V0 = V3 donc ce LFSR est de période 3 donc on en déduit que :
V0 = V3 = V6 = V9 = V12 = V15 = V18 [1;0;0;1;0;0;1;0;0;1]
V1 = V4 = V7 = V10 = V13 = V16 = V19 = [0;1;0;0;1;0;0;1;0;0]
V2 = V5 = V8 = V11 = V14 = V17 = V20 = [0;0;1;0;0;1;0;0;1;0]

Pour le deuxième :

V0 = [0;0;1]
V1 = [1;0;0]
V2 = [0;1;0]
V3 = [0;0;1]
Pareil on remarque que V0 = V3 donc ce LFSR est de période 3 donc on en déduit que :
V0 = V3 = V6 = V9 = V12 = V15 = V18 [0;0;1]
V1 = V4 = V7 = V10 = V13 = V16 = V19 = [1;0;0]
V2 = V5 = V8 = V11 = V14 = V17 = V20 = [0;1;0]

Exercice 4 :

1) On a $S(X) = \sum_{i \geq 0} r_i X^i$,

$$R(X) = \sum_{i=0}^{\ell} \alpha_i X^i \quad \text{et} \quad G(X) = \sum_{i=0}^{\ell-1} \left(\sum_{j=0}^{i} \alpha_{i-j} r_j \right) X^i$$

On cherche a prouver que $S(X)R(X) = G(X)$

$$S(X)R(X) = \left(\sum_{i \geq 0} r_i X^i \right) \left(\sum_{j=0}^{\ell} \alpha_j X^j \right)$$

$$= (r_0 + r_1 X + r_2 X^2 + \dots)(\alpha_0 + \alpha_1 X + \alpha_2 X^2 + \dots + \alpha_\ell X^\ell)$$

On multiplie et regroupe par degres de X donc

$$= r_0 \alpha_0 + r_1 X \alpha_0 + r_0 \alpha_1 X + \dots + r_\ell X^\ell \alpha_0 + \dots + r_0 \alpha_\ell X^\ell$$

On factorise

$$= r_0 \alpha_0 + (r_1 \alpha_0 + r_0 \alpha_1) X + \dots + (r_\ell \alpha_0 + \dots + r_0 \alpha_\ell) X^\ell$$

On garde uniquement $\ell - 1$ car ℓ est la taille du LFSR
et les valeur ensuite ne nous interesse pas (LFSR commence a l'indice 0)

$$= \sum_{i=0}^{\ell-1} \left(\sum_{j=0}^{i} r_j \alpha_{i-j} \right) X^i = G(X) \quad \square$$

2)

La fonction qui calcul le triplet est la fonction
getTriplet l avec l LFSR.
Qui renvoie le triplet (l,G(X) ,R(X)).

3)

La fonction qui calcul le triplet est la fonction
tripletToLFSR t avec t triplet (l,G(X) ,R(X)).
Qui renvoie le LFSR associé au triplet t.

Exercice 4 :

4) LFSR 1 : $\begin{cases} r_0 = r_3 = r_6 = r_9 = 1, \\ r_1 = r_2 = r_4 = r_5 = r_7 = r_8 = 0, \\ r_n = r_{n-1} \oplus r_{n-3} \oplus r_{n-4} \oplus r_{n-7} \oplus r_{n-10} \quad \text{si } n \geq 10 ; \end{cases}$

Donc $R(x) = 1 + X + X^3 + X^4 + X^7 + X^{10}$

On calcul $G(x)$:

i	j	$\alpha_{i-j} r_j = a$	$\sum_{j=0}^{i} a$
0	0	$1 \otimes 1 = 1$	1
1	0	$1 \otimes 1 = 1$	$1 \oplus 0 = 1$
1	1	$1 \otimes 0 = 0$	
2	0	$0 \otimes 1 = 0$	$0 \oplus 0 \oplus 0 = 0$
2	1	$1 \otimes 0 = 0$	
2	2	$1 \otimes 0 = 0$	
.	.	.	
.	.	.	
.	.	.	

Donc $G(x) = 1 + X + X^7$

On va donc chercher un facteur commun $T(x)$ aux polynomes $G(x)$ et $R(x)$

Donc on calcul $T(x) = PGCD(G(x), R(x))$

$$\begin{array}{l|l} X^{10} + X^7 + X^4 + X^3 + X + 1 & X^7 + X + 1 \\ \hline -(X^{10} + X^4 + X^3) & X^3 + 1 \\ \hline \quad X^7 + X + 1 & \\ \quad -(X^7 + X + 1) & \\ \hline \qquad 0 & \end{array}$$

reste de $G(x)$ par $R(x) = 0$
donc $PGCD(G(x), R(x)) = X^7 + X + 1$

le LFSR $(l, G(x), R(x))$ genere le meme flux qu le LFSR $(l', G(x)/T(x), R(x)/T(x))$

On a $G(x)/T(x) = 1$ et $R(x)/T(x) = X^3 + 1$

LFSR 2 : $\begin{cases} r_0 = 1 \\ r_1 = r_2 = 0 \\ r_n = r_{n-3} \end{cases}$

On a $R'(x) = 1 + X^3$
et $G'(x) = 1$

On en conclus que les deux LFSR produise le meme flux de valeur car

$G'(x) = G(x)/T(x)$ et $R'(x) = R(x)/T(x)$ □

5)

La fonction qui calcul le plus petit triplet est la fonction « smallest_triplet t » avec t triplet (l,G(X) ,R(X)).
Qui renvoie le plus petit triplet associé au triplet t.

III. Application à la cryptographie

Exercice 6 :

La fonction qui chiffre/Déchiffre un mot « mot » avec un LFSR « lfsr » est la fonction « chiffrage mot lfsr »