Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут імені Ігоря Сікорського”

Лабораторна робота

із Криптографії №4

Побудова регістрів зсуву з лінійним зворотним зв’язком та дослідження їх властивостей

Виконали:

студентки групи ФБ - 74

Горобець Ангеліна

Пудім Єлизавета

Перевірено_____

Київ 2019

КРИПТОГРАФІЯ
КОМП'ЮТЕРНИЙ ПРАКТИКУМ №4
Побудова регістрів зсуву з лінійним зворотним зв'язком та дослідження їх властивостей

Мета роботи
Ознайомлення з принципами побудови регістрів зсуву з лінійним зворотним зв'язком; практичне освоєння їх програмної реалізації; дослідження властивостей лінійних рекурентних послідовностей та їх залежності від властивостей характеристичного полінома регістра.

Порядок виконання роботи
0. Уважно прочитати методичні вказівки до виконання комп'ютерного практикуму. 1. Вибрати свій варіант завдання згідно зі списком. Варіанти завдань містяться у файлі Crypto_CP4 LFSR_Var.
2. За даними характеристичними многочленами $p_1(x)$, $p_2(x)$ скласти лінійні рекурентні співвідношення для ЛРЗ, що задаються цими характеристичними многочленами.
3. Написати програми роботи кожного з ЛРЗ L_1 , L_2 .
4. За допомогою цих програм згенерувати імпульсні функції для кожного з ЛРЗ і підрахувати їх періоди.
5. За отриманими результатами зробити висновки щодо влавствостей кожного з характеристичних многочленів $p_1(x)$, $p_2(x)$: многочлен примітивний над F_2 ; не примітивний, але може бути незвідним; звідний.
6. Для кожної з двох імпульсних функцій обчислити розподіл k-грам на періоді, $k \le n_i$, де n_i - степінь полінома $f_i(x)$, $i=1,2$ а також значення функції автокореляції $A(d)$ для $0 \le d \le 10$. За результатами зробити висновки.

Варіант 12:

$P_1(X)= X_{21}+X_{20}+X_{19}+X_{18}+X_{12}+X_{11}+X_{10}+X_8+X_6+X_5+1$
$P_2(X)= X_{23}+X_{22}+X_{21}+X_{20}+X_{19}+X_{16}+X_{15}+X_{13}+X_{12}+X_9+X_6+X_3+1$

Довжини періодів:
L_1: $2^{21}-1 = 2097151 \Rightarrow P_1(x)$ – примітивний поліном поля F_2
L_2: $94185 \Rightarrow P_2(x)$ – не примітивний та звідний

Значення автокореляції:

L_1:	L_2:
d= 0 0	d = 1 : 0
d= 1 1048576	47092
d= 2 1048576	47092
d= 3 1048576	47092
d= 4 1048576	47088
d= 5 1048576	47092
d= 6 1048576	47092
d= 7 1048576	47096
d= 8 1048576	47096
d= 9 1048576	47092
d= 10 1048576	47092

Висновок:

В даному комп'ютерному практикумі було набуто навичок роботи з лінійними регістрами зсуву, а саме: їх програмна реалізація, дослідження властивостей характеристичного полінома регістра. Окрім цього було досліджено властивості лінійних рекурентних послідовностей

Програмна реалізація:

```
p2_state = [0]*20 + [1]
p2_instate = [0]*20 + [1]
p2 = []
i = 0
while p2 != p2_instate :
p2_state.append(p2_state[i] ^ p2_state[i+5] ^ p2_state[i+6] ^ p2_state[i+8] ^ p2_state[i+10] ^
p2_state[i+11] ^ p2_state[i+12] ^ p2_state[i+18] ^ p2_state[i+19] ^ p2_state[i+20])
p2=p2_state[-21:]
i += 1
print(len(p2_state)-len(p2_instate))
print(2**21 -1)
#2097151 - полином примитивный

p3_state = [0]*22+[1]
p3_instate = [0]*22+[1]
p3 = []
i = 0
while p3 != p3_instate :
p3_state.append(p3_state[i] ^ p3_state[i+3] ^ p3_state[i+6] ^ p3_state[i+9] ^ p3_state[i+12] ^
p3_state[i+13] ^ p3_state[i+15] ^ p3_state[i+16] ^ p3_state[i+19] ^ p3_state[i+20] ^ p3_state[i+21] ^
p3_state[i+22])
p3=p3_state[-23:]
i += 1
print(len(p3_state)-len(p3_instate))
#94185 - не делит 2^23-1 - полином приводимый
print((2**23 -1)/94185)
```

```
print(p2_state[:-21].count(1)) #1048576
print(p2_state[:-21].count(0)) #1048575
#единичек на 1 больше, чем 0


#Автокорреляция
def autocorr(s, d, T):
A = sum((s[i]+s[(i+d)%T])%2 for i in range(0, T))
return A
for d in range(0,11):
print('d =',d,autocorr(p2_state, d, 2097151))
for d in range(0,11):
print(autocorr(p3_state, d, 94185))


#k-грами
p2_state = ''.join(str(i) for i in p2_state[21:])
for k in range(2,6):
bigrams_step2 = [p2_state[i:i+k] for i in range(0, len(p2_state), k)]
from collections import Counter
res = Counter(bigrams_step2)
print(res)


p3_state = ''.join(str(i) for i in p3_state[23:])
for k in range(2,6):
ngrams= [p3_state[i:i+k] for i in range(0, len(p3_state), k)]
from collections import Counter
res = Counter(ngrams)
print(res)
```

Розподіл K-грам полінома P1:

1:1048576, 0:1048575
Counter({'10': 262804, '00': 261965, '11': 261965, '01': 261841})
Counter({'011': 87734, '101': 87521, '001': 87516, '000': 87442, '111': 87437, '010': 87409, '100': 87153, '110': 86838})
Counter({'1010': 33092, '1000': 32957, '0111': 32918, '0001': 32837, '0010': 32822, '1011': 32818, '1101': 32810, '1100': 32727, '0000': 32720, '1110': 32713, '0101': 32704, '1111': 32678, '1001': 32670, '0110': 32640, '0011': 32623, '0100': 32558})
Counter({'11101': 13260, '01001': 13244, '01110': 13235, '10000': 13227, '01101': 13216, '00100': 13203, '00011': 13199, '11111': 13191, '10001': 13170, '01010': 13162, '10100': 13158, '11001': 13134, '11100': 13132, '01000': 13131, '00111': 13116, '01100': 13107, '11011': 13099, '01111': 13099, '00010': 13096, '11000': 13089, '10011': 13078, '11110': 13057, '01011': 13049, '10010': 13044, '00001': 13036, '00101': 13020, '10110': 13016, '00000': 12996, '00110': 12989, '10101': 12983, '11010': 12950, '10111': 12944})

Розподіл K-грам полінома P2:

Counter({'01': 11952, '10': 11723, '00': 11709, '11': 11708})
Counter({'110': 3943, '100': 3927, '111': 3927, '010': 3927, '001': 3927, '000': 3922, '011': 3911, '101': 3911})
Counter({'0101': 1521, '0001': 1520, '0110': 1518, '1110': 1496, '1100': 1481, '1101': 1479, '0011': 1472, '1001': 1466, '0100': 1466, '1010': 1465, '0111': 1461, '1000': 1457, '0000': 1445, '1111': 1443, '1011': 1433, '0010': 1423})
Counter({'11111': 616, '10001': 612, '01110': 607, '01000': 607, '11001': 603, '01011': 603, '01101': 603, '00000': 600, '11100': 599, '00011': 599, '10010': 595, '00110': 595, '00101': 591, '10111': 591, '10100': 590, '01010': 588, '10101': 588, '11000': 588, '11010': 586, '00100': 584, '10110': 584, '00010': 584, '01100': 580, '01111': 579, '11011': 576, '10011': 575, '11101': 571, '10000': 571, '01001': 571, '11110': 567, '00001': 567, '00111': 567})