«Київський Політехнічний Інститут ім. Ігоря Сікорського»

Фізико-технічний інститут

КРИПТОГРАФІЯ

КОМП’ЮТЕРНИЙ ПРАКТИКУМ №4

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

Виконали

студенти групи ФБ-73:

Деркач Вячеслав

Михалко Дмитро

Перевірили:

Чорний О.М., Завадська Л.А.

Київ 2019

Мета комп'ютерного практикуму:

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.

Постановка задачі:

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 $_1L$, $_2L$.
4. За допомогою цих програм згенерувати імпульсні функції для кожного з ЛРЗ і підрахувати їх періоди.
5. За отриманими результатами зробити висновки щодо влавстивостей кожного з характеристичних многочленів $p_1(x)$, $p_2(x)$: многочлен примітивний над $_2F$; 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.

Хід роботи:

1) Ми прочитали завдання та методичні вказівки;
2) Проаналізували завдання та виписали всі нюанси та деталі лабораторної;
3) .За даними характеристичними многочленами p1(x), p2(x) склали лінійні рекурентні співвідношення для ЛРЗ, що задаються цими характеристичними многочленами.
4) Написали програми роботи кожного з ЛРЗ $_1L$, $_2L$.
5) За допомогою цих програм згенерувати імпульсні функції для кожного з ЛРЗ і підрахувати їх періоди.
6) За отриманими результатами зробити висновки щодо влавстивостей кожного з характеристичних многочленів $p_1(x)$, $p_2(x)$: многочлен примітивний над $_2F$; не примітивний, але може бути незвідним; звідний.
7) . Для кожної з двох імпульсних функцій обчислити розподіл k-грам на періоді, k≤ni, де ni - степінь полінома fi(x), i=1,2 а також значення функції автокореляції A(d) для 0≤ d ≤ 10. За результатами зробити висновки.
8) Створили протокол, що описує нашу роботу;
9) Відправили все на Github;

Варіант 8:

$P_1(X) = X_{20} + X_{19} + X_{17} + X_{15} + X_{14} + X_4 + 1$
$P_2(X) = X_{24} + X_{21} + X_{15} + X_{14} + X_{11} + X_9 + X_8 + X_5 + X_4 + X_3 + 1$

Підраховані довжини періодів імпульсних функцій 1 L , 2 L ;

T for 1 = 41943

T for 2 = 16777215

Обчислені розподіли k-грам, $k \leq ni$,i=1,2 на періодах імпульсних Функцій 1 L , 2 L ;

k-grams for 1

{'00': 0.25008340879843666, '01': 0.25465897716982033, '10': 0.24460225918688336, '11': 0.2506553548448596, '000': 0.126617573461071, '001': 0.1263315936226496, '010': 0.12297133052119825, '011': 0.12189890612711804, '100': 0.1246872095517266, '101': 0.12239937084435547, '110': 0.1269750482590972, '111': 0.1281189676127833, '0000': 0.06348903717826501, '0001': 0.06425166825548141, '0010': 0.0555767397521449, '0011': 0.06224976167778837, '0100': 0.06558627264061011, '0101': 0.06606291706387035, '0110': 0.061010486177311724, '0111': 0.06320305052430887, '1000': 0.06024785510009533, '1001': 0.061487130600571975, '1010': 0.06291706387035272, '1011': 0.06377502383222117, '1100': 0.0652049571020019, '1101': 0.06167778836987607, '1110': 0.06129647283126787, '1111': 0.06196377502383222, '00000': 0.03169685414680648, '00001': 0.027645376549094377, '00010': 0.0328884652049571, '00011': 0.03336510962821735, '00100': 0.0328884652049571, '00101': 0.033245948522402285, '00110': 0.03205433746425167, '00111': 0.02979027645376549, '01000': 0.03312678741658723, '01001': 0.02800285986653956, '01010': 0.03133937082936129, '01011': 0.026453765490943755, '01100': 0.02669208770257388, '01101': 0.034318398474737846, '01110': 0.031816015252621545, '01111': 0.03360343183984747, '10000': 0.03026692087702574, '10001': 0.03169685414680648, '10010': 0.030028598665395614, '10011': 0.03419923736892278, '10100': 0.028717826501429934, '10101': 0.03098188751191611, '10110': 0.035986653956148716, '10111': 0.028836987607244995, '11000': 0.030624404194470926, '11001': 0.03098188751191611, '11010': 0.03110104861773117, '11011': 0.03193517635843661, '11100': 0.034080076263107724, '11101': 0.029194470924690182, '11110': 0.032411820781696854, '11111': 0.030028598665395614}

k-grams for 2

{'00': 0.2500706016091564, '01': 0.24970892109893178, '10': 0.25015106777408774, '11': 0.2500694095178241, '000': 0.12512309087329565, '001': 0.12487793730541717, '010': 0.12487775849172798, '011': 0.12487775849172798, '100': 0.1250608637094578, '101': 0.1250608637094578, '110': 0.1250608637094578, '111': 0.1250608637094578, '0000': 0.06246845427935806, '0001': 0.0624787062660381, '0010': 0.06240527343121358, '0011': 0.06263296290282858, '0100': 0.0624217242935606 3, '0101': 0.06235305982463381, '0110': 0.06243602939125372, '0111': 0.06240074348361077, '1000': 0.0625807492962488, '1001': 0.06245701020120358, '1010': 0.0626482216737012, '1011': 0.0626420227980342, '1100': 0.06268469967281858, '1101': 0.06251756844810433, '1110': 0.06248466672341022, '1111': 0.06238810731398187, '00000': 0.031220579553186207, '00001': 0.031127596412638913, '00010': 0.0312804821533464 9, '00011': 0.03143336789405406, '00100': 0.0312804821533464 9, '00101': 0.0312804821533464 9, '00110': 0.031127894435525282, '00111': 0.031127894435525282, '01000': 0.03135677601225709, '01001': 0.03135677601225709, '01010': 0.031204188294435883, '01011': 0.031204188294435883, '01100': 0.03135677601225709, '01101': 0.031204188294435883, '01110': 0.03135677601225709, '01111': 0.031204188294435883, '10000': 0.0312804821533464 9, '10001': 0.0312804821533464 9, '10010': 0.031127894435525282, '10011': 0.031127894435525282, '10100': 0.0312804821533464 9, '10101': 0.031127894435525282, '10110': 0.0312804821533464 9, '10111': 0.031127894435525282, '11000': 0.031204188294435883, '11001': 0.03135677601225709, '11010': 0.031204188294435883, '11011': 0.03135677601225709, '11100': 0.031204188294435883, '11101': 0.031204188294435883, '11110': 0.03135677601225709, '11111': 0.03135677601225709}

Значення функцій автокореляції $Ad(s)$ для $0 \leq d \leq 10$, для відповідних імпульсних функцій;

A for 1
A(0) = 0
A(1) = 20990
A(2) = 20990
A(3) = 20990
A(4) = 20994
A(5) = 20990
A(6) = 20990
A(7) = 20990
A(8) = 20994
A(9) = 20992
A(10) = 20992
A for 2
A(0) = 0
A(1) = 8388606
A(2) = 8388608
A(3) = 8388610
A(4) = 8388608
A(5) = 8388610
A(6) = 8388610
A(7) = 8388606
A(8) = 8388608
A(9) = 8388608
A(10) = 8388608

Програмний код:

```
indexlist_test = [0,0,1,0,0,0,0,1,0,0,0,0,0]
indexlist_1 = [1,0,1,0,1,1,0,0,0,0,0,0,0,0,1,0,0,0,1]
indexlist_2 = [0,0,1,0,0,0,0,0,1,1,0,0,1,0,1,1,0,0,1,1,1,0,0,1]
def createIndexListLRZ(indexlist):
    indexlist_LRZ = []
    for i in range(len(indexlist)-1):
        indexlist_LRZ.append(0)
    indexlist_LRZ.append(1)
    return indexlist_LRZ

def createLRZ(indexlist):
    LRZ = createIndexListLRZ(indexlist)
    forcheck = list(LRZ)
    counter = 0
    while 1:
        forcheck_2 = list(LRZ)
        counter += 1
        forswap = indexlist[0]*LRZ[0]
        for i in range(1,len(indexlist)):
            forswap += indexlist[i]*LRZ[i]
            LRZ[i] = forcheck_2[i-1]
        LRZ[0] = forswap % 2
        if LRZ == forcheck:
            break
    return counter

def createLRZforA(indexlist):
    LRZ = createIndexListLRZ(indexlist)
    forcheck = list(LRZ)
    counter = 0
    listforA = list(LRZ)
    while 1:
        forcheck_2 = list(LRZ)
        counter += 1
        forswap = indexlist[0]*LRZ[0]
        for i in range(1,len(indexlist)):
            forswap += indexlist[i]*LRZ[i]
            LRZ[i] = forcheck_2[i-1]
        LRZ[0] = forswap % 2
        listforA.append(forswap % 2)
        if LRZ == forcheck:
            break
    return listforA

def autoCor(indexlist):
    T = createLRZ(indexlist)
    listforA = createLRZforA(indexlist)
    for i in range(11):
        A = 0
        for j in range(T):
```

```
            A += (listforA[j] + listforA[(j+i)%T]) % 2
        print('A(',i,') = ',A)

def dictForK_grams(indexlist):
    n = len(indexlist)
    dictK = {}
    for i in range(4,int('1'*6,2) + 1):
        dictK[str(bin(i)[3:])] = 0
    return dictK

def bigramCount(indexlist):
    text = createLRZforA(indexlist)
    k_gramDict = dictForK_grams(indexlist)
    for position in range(0,len(text)-1,2):
        k_gramDict[str(text[position])+str(text[position+1])] += 1
    for position in range(0,len(text)-2,3):
        k_gramDict[str(text[position])+str(text[position+1])+str(text[position+2])] += 1
    for position in range(0,len(text)-3,4):
        k_gramDict[str(text[position])+str(text[position+1])+str(text[position+2])+str(text[position+3])] += 1
    for position in range(0,len(text)-4,5):

k_gramDict[str(text[position])+str(text[position+1])+str(text[position+2])+str(text[position+3])+str(text[position+4]
)] += 1
    for elem in k_gramDict:
        k_gramDict[elem]/= len(text)//len(elem)
    print(k_gramDict)

print('T for 1 = ',createLRZ(indexlist_1))
print('A for 1')
autoCor(indexlist_1)
print('k-grams for 1')
bigramCount(indexlist_1)
print('T for 2 = ',createLRZ(indexlist_2))
print('A for 2')
autoCor(indexlist_2)
print('k-grams for 2')
bigramCount(indexlist_2)
```

Висновки:

Ми провели 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.