Міністерство освіти і науки України
НТУУ «Київський політехнічний інститут»
Фізико-технічний інститут

КРИПТОГРАФІЯ

КОМП’ЮТЕРНИЙ ПРАКТИКУМ №4

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

Виконали:
Студенти III курсу ФТІ
групи ФБ-71
Бабенко І.М.,Гончаренко Д.А.

Перевірив:
Чорний О.
Завадська Л.О.
Савчук М.М.

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 p1(x), p2(x) скласти лінійні рекурентні співвідношення для ЛРЗ, що задаються цими характеристичними многочленами.
3. Написати програми роботи кожного з ЛРЗ L1 , L2 .
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 p1(x), p2(x): многочлен примітивний над F2 ; не примітивний, але може бути незвідним; звідний.
6. Для кожної з двох імпульсних функцій обчислити розподіл k-грам на періоді, $k \leq ni$, де ni - степінь полінома fi(x), i=1,2 а також значення функції автокореляції A(d) для $0 \leq d \leq 10$. За результатами зробити висновки.

Результати:

$$P_1(X)= X^{20}+X^{17}+X^{15}+X^{14}+X^9+X^7+X^5+X^3+X^2+X+1$$

Period = 1048575

2-граммы	**3-граммы**	**4-граммы**	**5-граммы**	**Autocor**
01:130644	011:43904	0101:16347	11011:6496	1: 524288
00:131111	100:43648	1000:16528	01000:6624	2: 524288
11:131111	101:43904	1010:16462	10100:6560	3: 524288
10:131421	010:43648	1101:16461	00001:6560	4: 524288
	110:43392	1100:16206	00101:6368	5: 524288
	000:43733	1111:16302	10001:6560	6: 524288
	111:43648	0000:16374	01010:6560	7: 524288
	001:43648	1011:16580	10101:6624	8: 524288
		0110:16061	10010:6560	9: 524288
		0011:16470	11110:6624	10: 524288
		1110:16556	00010:6560	
		0100:16373	00111:6560	
		0001:16417	10000:6496	
		0111:16234	01111:6560	
		0010:16369	01001:6688	
		1001:16403	11100:6688	
			01011:6496	
			00011:6496	
			01101:6496	
			11101:6496	
			11111:6560	
			11001:6432	
			10111:6560	
			01100:6432	
			10011:6496	
			11010:6560	
			00000:6547	
			01110:6624	
			00110:6624	
			10110:6624	
			11000:6624	
			00100:6560	

$$P_2(X)= X^{24}+X^{22}+X^{18}+X^{17}+X^{16}+X^{15}+X^{12}+X^{11}+X^9+X^4+X^2+X+1$$

Period = 1118481

2-граммы	3-граммы	4-граммы	5-граммы	Autocor
01:139828	000:46659	0010:17680	01100:7093	1: 559680
10:139969	101:46536	1101:17614	00010:6960	2: 559392
00:140126	111:46120	0100:17497	01110:6954	3: 559392
11:139317	110:46744	0000:17591	11110:6980	4: 558432
	001:46744	0101:17467	01011:6971	5: 560000
	100:46744	0110:17471	00101:6908	6: 559488
	011:46744	1000:17482	00001:6936	7: 559488
	010:46536	1110:17374	10101:6988	8: 559392
		1111:17477	10001:7207	9: 558432
		1100:17219	10111:7089	10: 559488
		0111:17285	11011:7008	
		0001:17697	00110:7062	
		1011:17502	00011:7078	
		1001:17330	10110:6957	
		0011:17369	00111:6905	
		1010:17565	10010:6972	
			00100:6949	
			10011:6990	
			00000:6984	
			01111:6817	
			10100:7019	
			11001:6946	
			01001:6987	
			01000:7136	
			01101:6986	
			11000:6944	
			10000:7054	
			11101:6964	
			11100:6855	
			01010:7061	
			11010:7017	
			11111:6919	

Код:

```
import AppKit
import Darwin.C.math

// ------------------------------------------------ SOURCE -
------------------------------------------------ //
var bigram : [String : Int] = ["00":0, "01":0, "10":0, "
11":0]

var threegram : [String : Int] = ["000":0, "001":0, "01
0":0, "011":0, "100":0, "101":0, "110":0, "111":0]

var fourgram : [String : Int] = ["0000":0, "0001":0, "0
010":0, "0011":0, "0100":0, "0101":0, "0110":0, "011
1":0, "1000":0, "1001":0, "1010":0, "1011":0, "1100"
:0, "1101":0, "1110":0, "1111":0]

var fivegram : [String : Int] = ["00000":0, "00001":0,
"00010":0, "00011":0, "00100":0, "00101":0, "00110
":0, "00111":0, "01000":0, "01001":0, "01010":0, "01
011":0, "01100":0, "01101":0, "01110":0, "01111":0,
"10000":0, "10001":0, "10010":0,  "10011":0, "10100
":0, "10101":0, "10110":0, "10111":0, "11000":0, "11
001":0, "11010":0, "11011":0, "11100":0, "11101":0,
"11110":0, "11111":0]

let polinom1 : [Int] = [1,1,1,1,0,1,0,1,0,1,0,0,0,0,1,1,0
,1,0,0] // first polinom
```

```
let impulse1 : [Int] = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,1] //impulse for first polinom
let polinom2 : [Int] = [1,1,1,0,1,0,0,0,0,1,0,1,1,0,0,1,1
,1,1,0,0,0,1,0] // second polinom
let impulse2 : [Int] = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,1] //impulse for second polinom
var period = ""

// ------------------------------------------------
- FUNCTIONS ----------------------------------------------
--- //
func period_count(){
    var temp = [Int]()
    for i in 0...(impulse2.count)-1{
        let a : Int = impulse2[i]
        temp.append(a)
    }
    repeat {
        var sum_array = [Int]()
        for i in 0...(impulse2.count)-1{
            let a = polinom2[i] * temp[i]
            sum_array.append(a)
        }
        let sum : Int = (sum_array.reduce(0, +))%2
        temp.append(sum)
        period = period + String(temp[0])
        temp.removeFirst()
    } while(temp != impulse2)
    print("Period = \(period.count)")
}

func ngramm_count (dict: [String:Int], n: Int) -
> [String:Int] {
    var ngram = dict
    var temp_ngram = ""
    for character in period {
        temp_ngram = temp_ngram + String(character)
        if temp_ngram.count == n {
            ngram[temp_ngram]!+=1
            temp_ngram.removeAll()
        }}
    return ngram
}

func autocor (p: String, n: Int) -> Int{
    var period_int = period.compactMap{$0.wholeNu
mberValue}
    for i in 1...n{
        period_int.append(period_int[i-1])
    }
    var sum_array = [Int]()
    for i in 0...(period.count)-1 {
        let a = (period_int[i] + period_int[i+n])%2
        sum_array.append(a)
    }
    let sum : Int = sum_array.reduce(0, +)
    return sum
}

// ------------------------------------------------ MAIN -----
------------------------------------------- //
let methodStart = Date()
period_count()
print("\n")
print("2-граммы: ")
bigram = ngramm_count(dict: bigram, n: 2)
for item in bigram {
    print("\(item.key):\(item.value)")
}
print("\n")
print("3-граммы: ")
threegram = ngramm_count(dict: threegram, n: 3)
for item in threegram {
    print("\(item.key):\(item.value)")
}
print("\n")
print("4-граммы: ")
fourgram = ngramm_count(dict: fourgram, n: 4)
for item in fourgram {
    print("\(item.key):\(item.value)")
}
print("\n")
print("5-граммы: ")
fivegram = ngramm_count(dict: fivegram, n: 5)
for item in fivegram {
    print("\(item.key):\(item.value)")
}
print("\n")
for i in 1...10{
    print("Autocor \(i): \(autocor(p: period, n: i))")
}
print("\n")
let methodFinish = Date()
let executionTime = methodFinish.timeIntervalSince(
methodStart)
print("Execution time: \(executionTime)")
```

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.