Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Звіт
до ЛАБОРАТОРНОЇ РОБОТА №4
з дисципліни: “Спеціальні розділи обчислювальної математики”

Виконала студентка:
групи ФІ-03
Швець Катерина

Реалізація операцій у скінченних полях характеристики 2 (нормальний базис)
Варіант 18

1. Мета роботи

Одержання практичних навичок програмної реалізації обчислень у полі Галуа характеристики 2 в нормальному базисі; ознайомлення з прийомами ефективної реалізації критичних по часу ділянок програмного коду та методами оцінки їх ефективності.

m=641

2. Завдання

А) Реалізувати поле Галуа характеристики 2 степеня m в поліноміальному базисі з операціями:

1) знаходження константи 0 – нейтрального елемента по операції «+»;
2) знаходження константи 1 – нейтрального елемента по операції «»;
3) додавання елементів;
4) множення елементів;
5) обчислення сліду елементу;
6) піднесення елемента поля до квадрату;
7) піднесення елемента поля до довільного степеня (не вище $2^m - 1$, де m –розмірність розширення);
8) знаходження оберненого елемента за множенням;
9) конвертування (переведення) елемента поля в m-бітний рядок (строкове зображення) і навпаки, де m – розмірність розширення;

Вхідні дані:

```
a=ONB("1110110010111110000010010011000101011011100101010000011100001110
1101011100010010101001001010101000111110001001111000101001010001011011
0100100100010100110110000111100010101110000000000011100001011000001111
0001110111001100101100011110")
b=ONB("0000110000000100100100101101011010110111010100100010111010101001
0111100110100111101000110000001000001100100100010111010010011001001000
1110110000011011011101101110010001110010000011001001101001100110100110
0100111011101110110011001")
c=ONB("1110110010111110000010010011000101011011100 1")
d=ONB("0000110000000100100100101101011010110111010 1")
m=ONB("1010100010111")
```

Результат роботи:

```
Сума:  1110000010111010100110111110011111101100110001110010100110100111101011101011010100000111101010000011001010110110111000110
1100011001010010100101000011111010111010011100110111000000110010100010001111101010010010100101011101111000010111
Слід:  0
Множення:  0110101110110000111000100001000010101011000111001011110010001000100000100010100110001110111000100001010100011011111110
1110110111001111011100011000000110100011001101101110111100100100110010100000010000111011000001001011011000010101001110010001100
1101010001010010100000000111110011110000101010000100100000000011000110011000000110101111110100011100010000001110011000000010000000
1001000111011110100100001010001110100100010101000000010100110100101100100110101100011010001111000000110110100101111011001011001110
00110001
Квадрат числа:  000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000111011001011111000001001001100010101101110010101000001110000111011010111000100101010010010101
0100011111000100111100010100101000101101100100100100010100110110000011110001010111000000000000111000010110000011111000111011100110
0101100011100
```

Реалізовані операції

Додавання (побітовий XOR)

```
    def __add__(self, other):
        res = ""
        for i in range(len(self.n)):
            if self.n[i] == other.n[i]: res=res+ "0"
            else: res =res+ "1"
        res=ONB(res)
        return res
```

Множення(за допомогою мультиплікативної матриці)

```
    def mul(self, other, matrix):
        result = ''
        for i in range(self.m):
            ao = self << i
            bo = other << i
            string = ''
            for i in range(self.m):
                res = 0
                for j in range(self.m):
                    a = int(ao.n[j])
                    b = int(matrix[i][j])
                    res += a*b
                string += str(res % 2)
            res = 0
            for i in range(self.m):
                a = int(string[i])
                b = int(bo.n[i])
                res += a * b
            result =result+ str(res % 2)
        return ONB(result)
```

Обчислення сліду (сума коефіцієнтів за порядку поля)

```
    def trace(self):
        res=0
        for i in range(len(self.n)):
            if self.n[i]=="1": res=res+1
        if res%2==0: return 0
        else: return 1
```

Квадрат:

```
    def powsq(self):
        return (self<<1).n
```

Код програми за посиланням: https://github.com/Kkaaats/Shvets-FI-03-Labs