СПЕЦІАЛЬНІ РОЗДІЛИ ОБЧИСЛЮВАЛЬНОЇ МАТЕМАТИКИ

Комп’ютерний практикум №3

Реалізація операцій у скінченних полях характеристики 2

(Поліноміальний базис)

ФБ-23 Моісеєнко Дмитро

Мета роботи:

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

Завдання до комп’ютерного практикуму:

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

2m □1, де m – розмірність розширення);

8) знаходження оберненого елемента за множенням;

Хзображення) і навпаки, де m – розмірність розширення;

Мова програмування, семантика функцій, спосіб реалізації можуть обиратись довільно.

Під час конвертування елементів поля у бітові рядки потрібно враховувати конвенції щодо зображень елементів поля (зокрема, порядок бітів).

Обираю довільний 3 варіант. m = 179, p(x) = x^179+x^4+x^2+x+1

Хід роботи:

Напишемо бібліотеку для роботи з елементами в поліноміальному базисі при визначеному довільній моій варіанті і проведемо тести для коректності роботи цієї бібліотеки

Реалізація результатів:

Код Python – main.py

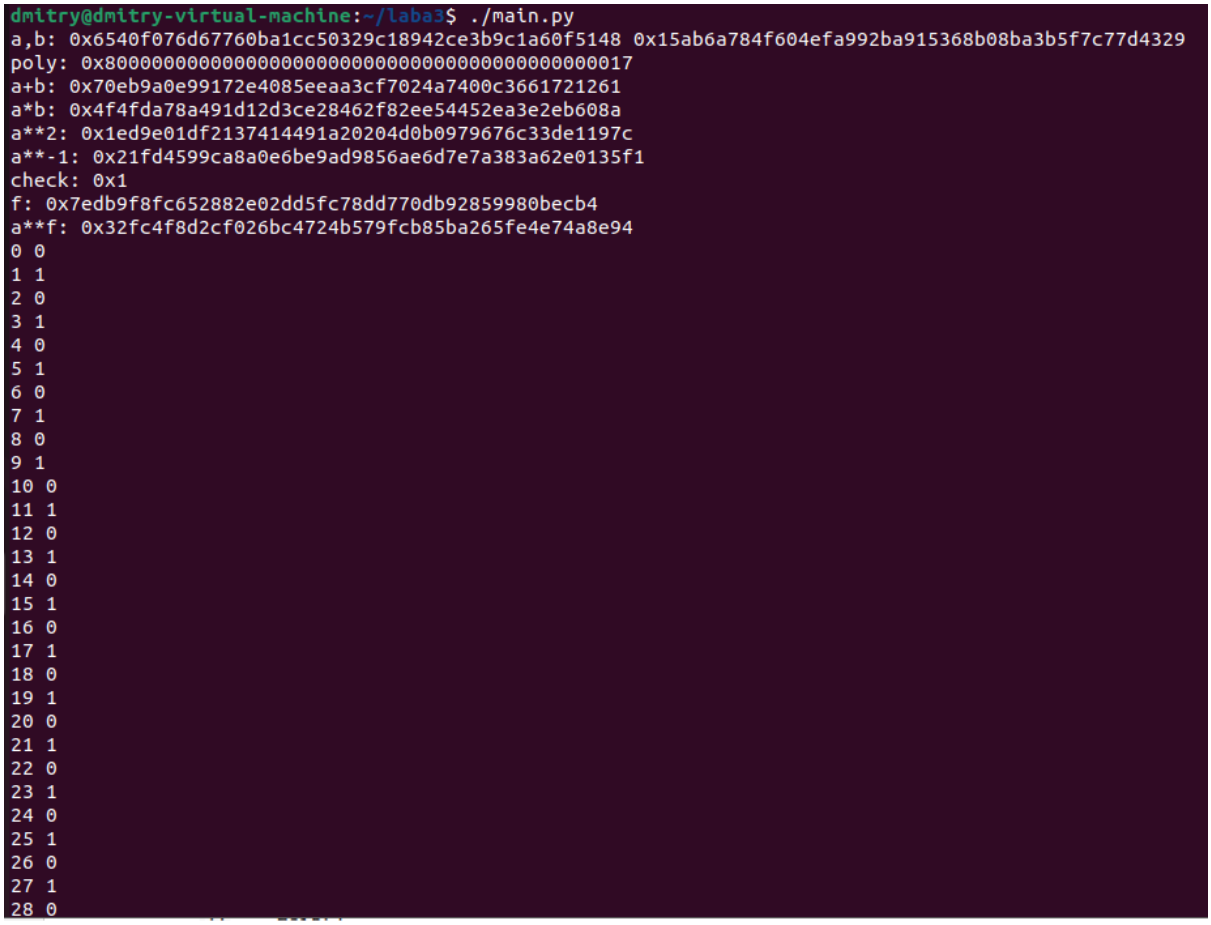
Open

main.py
~/laba3

```python
#!/usr/bin/python3
# from Mathe.poly import *
from Mathe.gf import *
from random import getrandbits

def main() -> None:
    fld = GF()
    BITS = fld.m
    A,B = getrandbits(BITS),getrandbits(BITS)
    a,b = fld(A),fld(B)
    print("a,b:", a,b)
    print("poly:", fld.poly)
    print("a+b:",a+b)
    print("a*b:",a*b)
    print("a**2:",a**2)
    a_ = a.inv()
    print("a**-1:",a_)
    print("check:",a*a_)
    f = fld(getrandbits(BITS))
    f.reduce()
    print("f:",f)
    print("a**f:",a**f)
    for i in range(50):
        f = fld(i)
        print(i,f.trace())
if __name__ == "__main__":
    main()

```


dmitry@dmitry-virtual-machine:~/laba3$ ./main.py
a,b: 0x6540f076d67760ba1cc50329c18942ce3b9c1a60f5148 0x15ab6a784f604efa992ba915368b08ba3b5f7c77d4329
poly: 0x80000000000000000000000000000000000000000000000017
a+b: 0x70eb9a0e99172e4085eeaa3cf7024a7400c3661721261
a*b: 0x4f4fda78a491d12d3ce28462f82ee54452ea3e2eb608a
a**2: 0x1ed9e01df2137414491a20204d0b0979676c33de1197c
a**-1: 0x21fd4599ca8a0e6be9ad9856ae6d7e7a383a62e0135f1
check: 0x1
f: 0x7edb9f8fc652882e02dd5fc78dd770db92859980becb4
a**f: 0x32fc4f8d2cf026bc4724b579fcb85ba265fe4e74a8e94
0 0
1 1
2 0
3 1
4 0
5 1
6 0
7 1
8 0
9 1
10 0
11 1
12 0
13 1
14 0
15 1
16 0
17 1
18 0
19 1
20 0
21 1
22 0
23 1
24 0
25 1
26 0
27 1
28 0

Ipython3:

```
[1]+  Stopped                 ./main.py
dmitry@dmitry-virtual-machine:~/laba3$ ipython3
Python 3.10.12 (main, Nov  6 2024, 20:22:13) [GCC 11.4.0]
Type 'copyright', 'credits' or 'license' for more information
IPython 7.31.1 -- An enhanced Interactive Python. Type '?' for help.

In [1]: from Mathe.gf import *

In [2]: from random import getrandbits

In [3]: fld = GF()

In [4]: BITS = fld.m

In [5]: A,B,C = getrandbits(BITS),getrandbits(BITS),getrandbits(BITS)

In [6]: f = fld(2**BITS - 1)

In [7]: a,b,c = fld(A),fld(B),fld(C)

In [8]: a**f
Out[8]: 0x1

In [9]: b*c + a*c
Out[9]: 0x36ee5c198502198de959df4de98e291cf5fb3e9af40d

In [10]: (a+b)*c
Out[10]: 0x36ee5c198502198de959df4de98e291cf5fb3e9af40d


[2]+  Stopped                 ipython3
```

Результати проведення Test3.py:

Додавання:

```
dmitry@dmitry-virtual-machine:~/laba3$ ./Test3.py
         19 function calls in 0.000 seconds

   Ordered by: standard name

   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
        1    0.000    0.000    0.000    0.000 <string>:1(<module>)
        1    0.000    0.000    0.000    0.000 Test3.py:10(add)
        1    0.000    0.000    0.000    0.000 gf.py:28(__init__)
        1    0.000    0.000    0.000    0.000 gf.py:46(__add__)
        1    0.000    0.000    0.000    0.000 gf.py:65(bitLen)
        1    0.000    0.000    0.000    0.000 {built-in method builtins.exec}
        2    0.000    0.000    0.000    0.000 {built-in method builtins.isinstance}
        9    0.000    0.000    0.000    0.000 {built-in method builtins.len}
        1    0.000    0.000    0.000    0.000 {built-in method builtins.max}
        1    0.000    0.000    0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}


         2774 function calls in 0.010 seconds
```

Множення:

```
Ordered by: standard name

ncalls  tottime  percall  cumtime  percall filename:lineno(function)
     1    0.000    0.000    0.010    0.010 <string>:1(<module>)
     1    0.000    0.000    0.010    0.010 Test3.py:12(mul)
     1    0.000    0.000    0.005    0.005 gf.py:111(reduce)
   178    0.000    0.000    0.002    0.000 gf.py:28(__init__)
    88    0.001    0.000    0.001    0.000 gf.py:46(__add__)
     9    0.005    0.001    0.005    0.001 gf.py:56(mulStep)
   416    0.003    0.000    0.003    0.000 gf.py:65(bitLen)
     1    0.000    0.000    0.010    0.010 gf.py:77(__mul__)
    88    0.001    0.000    0.002    0.000 gf.py:92(lshift)
     1    0.000    0.000    0.010    0.010 {built-in method builtins.exec}
   356    0.000    0.000    0.000    0.000 {built-in method builtins.isinstance}
  1541    0.000    0.000    0.000    0.000 {built-in method builtins.len}
    89    0.000    0.000    0.000    0.000 {built-in method builtins.max}
     1    0.000    0.000    0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}
     3    0.000    0.000    0.000    0.000 {method 'pop' of 'list' objects}


      506451 function calls (506273 primitive calls) in 1.676 seconds
```

Піднесення до степеня:

```
Ordered by: standard name

ncalls  tottime  percall  cumtime  percall filename:lineno(function)
     1    0.000    0.000    1.676    1.676 <string>:1(<module>)
     1    0.000    0.000    1.676    1.676 Test3.py:14(pow)
   269    0.051    0.000    1.102    0.004 gf.py:111(reduce)
 179/1    0.021    0.000    1.676    1.676 gf.py:127(__pow__)
 32893    0.072    0.000    0.384    0.000 gf.py:28(__init__)
 16177    0.133    0.000    0.343    0.000 gf.py:46(__add__)
   819    0.542    0.001    0.542    0.001 gf.py:56(mulStep)
 76001    0.631    0.000    0.648    0.000 gf.py:65(bitLen)
    91    0.005    0.000    1.077    0.012 gf.py:77(__mul__)
 16177    0.140    0.000    0.585    0.000 gf.py:92(lshift)
     1    0.000    0.000    1.676    1.676 {built-in method builtins.exec}
 65787    0.016    0.000    0.016    0.000 {built-in method builtins.isinstance}
280979    0.055    0.000    0.055    0.000 {built-in method builtins.len}
 16268    0.011    0.000    0.011    0.000 {built-in method builtins.max}
     1    0.000    0.000    0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}
   807    0.001    0.000    0.001    0.000 {method 'pop' of 'list' objects}
```

Inv:

```
      19721 function calls in 0.223 seconds

Ordered by: standard name

ncalls  tottime  percall  cumtime  percall filename:lineno(function)
     1    0.000    0.000    0.223    0.223 <string>:1(<module>)
     1    0.000    0.000    0.223    0.223 Test3.py:16(inv)
   276    0.000    0.000    0.002    0.000 gf.py:111(reduce)
    92    0.001    0.000    0.013    0.000 gf.py:166(__truediv__)
    93    0.000    0.000    0.000    0.000 gf.py:178(isnull)
     1    0.001    0.001    0.223    0.223 gf.py:183(inv)
  1113    0.004    0.000    0.013    0.000 gf.py:28(__init__)
   461    0.004    0.000    0.010    0.000 gf.py:46(__add__)
  2484    0.187    0.000    0.187    0.000 gf.py:56(mulStep)
  1894    0.013    0.000    0.013    0.000 gf.py:65(bitLen)
   276    0.009    0.000    0.203    0.001 gf.py:77(__mul__)
   185    0.001    0.000    0.004    0.000 gf.py:92(lshift)
     1    0.000    0.000    0.223    0.223 {built-in method builtins.exec}
  2226    0.001    0.000    0.001    0.000 {built-in method builtins.isinstance}
  9051    0.002    0.000    0.002    0.000 {built-in method builtins.len}
   737    0.001    0.000    0.001    0.000 {built-in method builtins.max}
     1    0.000    0.000    0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}
   828    0.000    0.000    0.000    0.000 {method 'pop' of 'list' objects}
```

Trace:

```
      256753 function calls in 0.643 seconds

Ordered by: standard name

ncalls  tottime  percall  cumtime  percall filename:lineno(function)
     1    0.000    0.000    0.643    0.643 <string>:1(<module>)
     1    0.000    0.000    0.643    0.643 Test3.py:18(trace)
   180    0.024    0.000    0.612    0.003 gf.py:111(reduce)
   179    0.023    0.000    0.639    0.004 gf.py:127(__pow__)
     1    0.001    0.001    0.643    0.643 gf.py:153(trace)
 16751    0.030    0.000    0.211    0.000 gf.py:28(__init__)
  8285    0.077    0.000    0.183    0.000 gf.py:46(__add__)
 38485    0.353    0.000    0.362    0.000 gf.py:65(bitLen)
  8106    0.090    0.000    0.333    0.000 gf.py:92(lshift)
     1    0.000    0.000    0.643    0.643 {built-in method builtins.exec}
 33502    0.011    0.000    0.011    0.000 {built-in method builtins.isinstance}
142438    0.029    0.000    0.029    0.000 {built-in method builtins.len}
  8285    0.004    0.000    0.004    0.000 {built-in method builtins.max}
     1    0.000    0.000    0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}
   537    0.000    0.000    0.000    0.000 {method 'pop' of 'list' objects}
```