Лабораторная работа №7

Дискретное логарифмирование в конечном поле

Выполнила:

Манаева Варвара Евгеньевна, НФИмд-01-24, 1132249514

Для проверки будем использовать число 221

1. Алгоритм, реализующий p-метод Полларда для задач дискретного логарифмирования

In [119…
```
function searching_for_gamma(a_diff, b_diff, p)
    for i in 1:p
        if b_diff*i % p == a_diff
            return i
        end
    end
    return "Not found"
end
```

Out[119… searching_for_gamma (generic function with 1 method)

In [89]:
```
function new_xab(x, a, b, p, alph, bett)
    if x % 3 == 0
        return x^2 % p, a*2 % (p-1), b*2 % (p-1)
    elseif x % 3 == 1
        return x*alph % p, (a+1) % (p-1), b
    else
        return x*bett % p, a, (b+1) % (p-1)
    end
end
```

Out[89]: new_xab (generic function with 1 method)

In [94]:
```
function metodPollarda(p, alp, bet)# , any_func::Function)
    if p % 2 == 0
        return "Incorrect input: p must be simple"
    end
    a_i = 0; b_i = 0; x_i = 1
    a_2i = 0; b_2i = 0; x_2i = 1
    i = 1
    tries = 1000
    data = zeros(Int64, (3, tries))
    data2 = zeros(Int64, (3, tries))
    while i <= tries
        x_i, a_i, b_i = new_xab(x_i, a_i, b_i, p, alp, bet)
        data[:, i] = [x_i, a_i, b_i]
```

```
        x_2i, a_2i, b_2i = new_xab(x_2i, a_2i, b_2i, p, alp, bet)
        x_2i, a_2i, b_2i = new_xab(x_2i, a_2i, b_2i, p, alp, bet)
        data2[:, i] = [x_2i, a_2i, b_2i]

        if x_i == x_2i
            display(data[:, 1:i])
            display(data2[:, 1:i])
            r = b_2i - b_i
            if r == 0
                return "Не найдено"
            else
                return searching_for_gamma(a_i - a_2i, r, p)
            end
        end
        i += 1
    end
    return "Делитель не найден"
end
```

Out[94]: metodPollarda (generic function with 1 method)

In [120…

```
p = 107
alp = 10
bet = 64
metodPollarda(p, alp, bet)
```

```
3×14 Matrix{Int64}:
 10  100  37  49  62  9  81  34  19  83  69  53  75  61
  1    2   3   4   5  5  10  20  21  22  22  44  44  88
  0    0   0   0   0  1   2   4   4   4   5  10  11  22
3×14 Matrix{Int64}:
 100  49  9  34  83  53  61  61  61  61  61  61  61  61
   2   4  5  20  22  44  88  72  40  82  60  16  34  70
   0   0  1   4   4  10  22  44  88  70  34  68  30  60
```

Out[120… 23

In [121…

```
p = 1019
alp = 2
bet = 5
metodPollarda(p, alp, bet)
```

```
3×51 Matrix{Int64}:
 2  10  20  100  200  1000  981  425  …   86  430  860  224  101  505  1010
 1   1   2    2    3     3    4    8     679  679  680  680  680  680   681
 0   1   1    2    2     3    3    6     374  375  375  376  377  378   378
3×51 Matrix{Int64}:
 10  100  1000  425  436  284  986  …  108  237  248   86  860  101  1010
  1    2     3    8   16   17   17     838  658  299  299  300  300   301
  1    2     3    6   14   15   17     102  205  410  412  413  415   416
```

Out[121… 10