Отчет по лабораторной работе 7

Дисциплина: Математические основы защиты информации и информационной безопасности

Дяченко З. К.

9 декабря 2022

Российский университет дружбы народов, Москва, Россия

Прагматика выполнения лабораторной работы

Данная лабораторная работа выполнялась мной для приобретения практических навыков реализации p-метод Полларда для задач дискретного логарифмирования.

Цель выполнения лабораторной работы

Ознакомится и реализовать программно алгоритм, реализующий p-метод Полларда для задач дискретного логарифмирования.

Задачи выполнения лабораторной работы

Реализовать программно алгоритм, реализующий p-метод Полларда для задач дискретного логарифмирования (рис. - fig. 1).

In [64]:
```
def polard (a, b, p, u, v):
    log_c=[]
    log_d=[]
    c=a**u*b**v%p
    c_107=a**u*b**v
    d=c
    d_107=c_107
    log_c.append([u, v])
    log_d.append([u, v])
    u_c=u
    u_d=u
    v_c=v
    v_d=v
    k=0
    i=0
    while (i==0):
        if (c<53):
            u_c+=1
            c=(c*a)%p
            log_c.append([u_c, v_c])
        else:
            v_c+=1
            c=(c*b)%p
            log_c.append([u_c, v_c])
        if (d<53):
            u_d+=1
            d=(d*a)%p
            if (d<53):
                u_d+=1
                d=(d*a)%p
                log_d.append([u_d, v_d])
            else:
                v_d+=1
                d=(d*b)%p
                log_d.append([u_d, v_d])
```

Задачи выполнения лабораторной работы

```
            log_d.append([u_d, v_d])
        else:
            v_d+=1
            d=(d*b)%p
            if (d<53):
                u_d+=1
                d=(d*a)%p
                log_d.append([u_d, v_d])
            else:
                v_d+=1
                d=(d*b)%p
                log_d.append([u_d, v_d])
        k+=1
        print(c, log_c[k], d, log_d[k])
        if (c==d):
            return(log_c[k], log_d[k])
```

```
In [65]: log_c, log_d= polard(10, 64, 107, 2, 2)
```

```
40 [3, 2] 79 [4, 2]
79 [4, 2] 56 [5, 3]
27 [4, 3] 75 [5, 5]
56 [5, 3] 3 [5, 7]
53 [5, 4] 86 [7, 7]
75 [5, 5] 42 [8, 8]
92 [5, 6] 23 [9, 9]
3 [5, 7] 53 [11, 9]
30 [6, 7] 92 [11, 11]
86 [7, 7] 30 [12, 12]
47 [7, 8] 47 [13, 13]
```

Figure 2: Реализация шагов 1 и 2 алгоритма, реализующего p-метод Полларда для задач дискретного логарифмирования

Задачи выполнения лабораторной работы

```
In [66]: log_c
Out[66]: [7, 8]

In [67]: log_d
Out[67]: [13, 13]

In [109]: koef_1=log_c[0]-log_d[0]

In [110]: koef_2=log_c[1]-log_d[1]

In [111]: def poisk(c, k, dell):
              for i in range(1, 100):
                  b=dell*i
                  for j in range (1, 100):
                      a=c+k*j
                      if (a==b):
                          return j
                  b=dell*(-i)
                  for j in range (1, 100):
                      a=c+k*j
                      if (a==b):
                          return j

In [112]: q=poisk(koef_1, koef_2, 53)

In [113]: q
Out[113]: 20
```

Figure 3: Реализация 3 шага алгоритма

Результаты выполнения лабораторной работы

Результатом выполнения работы стала реализация алгоритма, реализующего p-метод Полларда для задач дискретного логарифмирования, что отражает проделанную мной работу и полученные новые знания.