

Презентация по лабораторной работе №7. Дискретное логарифмирование в конечном поле

Хитяев Евгений Анатольевич НПМмд-02-21

25 декабря, 2021, Москва, Россия

Российский Университет Дружбы Народов

Цели и задачи

Цель лабораторной работы

Изучение задачи дискретного логарифмирования.

Выполнение лабораторной работы

Задача дискретного логарифмирования

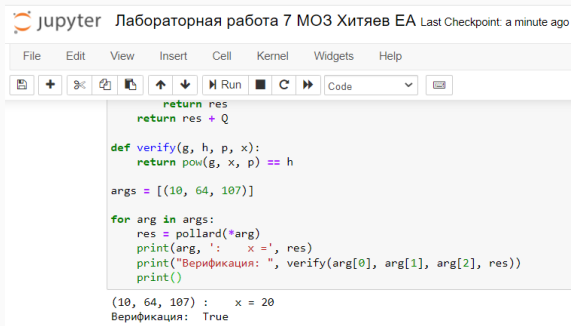
Решение задачи дискретного логарифмирования состоит в нахождении некоторого целого неотрицательного числа x , удовлетворяющего уравнению. Если оно разрешимо, у него должно быть хотя бы одно натуральное решение, не превышающее порядок группы.

р-алгоритм Полларда

- Вход. Простое число p , число a порядка r по модулю p , целое число b , $1 < b < p$; отображение f , обладающее сжимающими свойствами и сохраняющее вычислимость логарифма.
 - Выход. показатель x , для которого $a^x = b \pmod{p}$, если такой показатель существует.
1. Выбрать произвольные целые числа u, v и положить $c = a^u b^v \pmod{p}$, $d = c$
 2. Выполнять $c = f(c) \pmod{p}$, $d = f(f(d)) \pmod{p}$, вычисляя при этом логарифмы для c и d как линейные функции от x по модулю r , до получения равенства $c = d \pmod{p}$
 3. Приняв логарифмы для c и d , вычислить логарифм x решением сравнения по модулю r . Результат x или “Решения нет”.

Алгоритм полного перебора нашёл бы решение за число шагов не выше порядка данной группы.

Пример работы алгоритма



The image shows a Jupyter Notebook interface. The title bar reads "jupyter Лабораторная работа 7 МОЗ Хитяев ЕА Last Checkpoint: a minute ago". The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. The toolbar contains icons for file operations, a plus sign, undo, redo, copy, paste, up/down arrows, a run button, a stop button, a refresh button, a dropdown menu set to "Code", and a help icon. The code cell contains the following Python code:

```
        return res
    return res + Q

def verify(g, h, p, x):
    return pow(g, x, p) == h

args = [(10, 64, 107)]

for arg in args:
    res = pollard(*arg)
    print(arg, 'x =', res)
    print("Верификация: ", verify(arg[0], arg[1], arg[2], res))
    print()

(10, 64, 107) :    x = 20
Верификация:  True
```

Figure 1: Пример работы алгоритма

Получаем $x = 20$ для значений в данном примере.

Выводы

Результаты выполнения лабораторной работы

В ходе выполнения работы мне удалось изучить задачу дискретного логарифмирования, повторить р-алгоритм Полларда, а также реализовать алгоритм программно на языке Python.