Лабораторная работа 7

Серенко Данил Сергеевич, НФИмд-01-23

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

ПРЕЗЕНТАЦИЯ ПО ЛАБОРАТОРНОЙ РАБОТЕ №7

дисциплина: Математические основы защиты информации и информационной безопасности

Преподователь: Кулябов Дмитрий Сергеевич

Студент: Серенко Данил Сергеевич

Группа: НФИмд-01-23

МОСКВА

2023 г.

Прагматика выполнения лабораторной работы

Требуется реализовать:

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

Цель работы

Освоить на практике дискретное логарифмирование в конечном поле.

Выполнение лабораторной работы

1. Для реализации p-метода Полларда:

1. Функция, реализующая p-метод Полларда
2. Функция нахождения НОД
3. Расширенный алгоритм Евклида для вычисления модульного обратного элемента

```python
def pollard_p_method(p, a, b, f, r, u, v):
    # Выбор произвольных чисел u, v
    c = (a ** u * b ** v) % p
    d = c

    # Итерации метода Полларда
    while True:
        c = f(c) % p
        d = f(f(d)) % p

        # Если c = d, вычисляем логарифмы для c и d
        if c == d:
            # Вычисляем логарифм x решением сравнения по модулю r
            # Решения нет, если r и p не взаимно просты
            if gcd(r, p - 1) != 1:
                return "Решения нет"

            # Вычисляем логарифм x
            x = (u - v * modinv((u - v), r) * (c - a ** u) % r) % r
            return x


# Функция вычисления наибольшего общего делителя (GCD)
def gcd(a, b):
    while b:
        a, b = b, a % b
    return a
```

```python
def modinv(a, m):
    m0, x0, x1 = m, 0, 1
    while a > 1:
        q = a // m
        m, a = a % m, m
        x0, x1 = x1 - q * x0, x0
    return x1 + m0 if x1 < 0 else x1


# Пример использования
p = 107
a = 10
b = 64
r = 53
u = 2
v = 2


# Определение функции f
def f(c):
    if c < r:
        return (10 * c) % p
    else:
        return (64 * c) % p


result = pollard_p_method(p, a, b, f, r, u, v)
print("Решение:", result)
```

2. Основная фунция запуска где получаем входные значения и шифруем слово

```
C:\Users\Nitro\AppData\Local\Programs\Py
Решение: Решения нет

Process finished with exit code 0
```

output

Выводы

В результате выполнения работы я освоил на практике дискретное логарифмирование в конечном поле.