Цели и задачи

Цель лабораторной работы

Целью данной является изучение задачи дискретного логарифмирования.

Выполнение лабораторной работы

Задача дискретного логарифмирования

Пусть в некоторой конечной мультипликативной абелевой группе G задано уравнение g^{x}=a. (1) Решение задачи дискретного логарифмирования состоит в нахождении некоторого целого неотрицательного числа x, удовлетворяющего уравнению (1). Если оно разрешимо, у него должно быть хотя бы одно натуральное решение, не превышающее порядок группы. Это сразу даёт грубую оценку сложности алгоритма поиска решений сверху — алгоритм полного перебора нашёл бы решение за число шагов не выше порядка данной группы. Чаще всего рассматривается случай, когда G = ⟨g⟩, то есть группа является циклической, порождённой элементом g. В этом случае уравнение всегда имеет решение. В случае же произвольной группы вопрос о разрешимости задачи дискретного логарифмирования, то есть вопрос о существовании решений уравнения (1), требует отдельного рассмотрения.

p-алгоритм Полларда

- Вход. Простое число p, число a порядка r по модулю p, целое число bб $1 < b < p$; отображение f, обладающее сжимающими свойствами и сохраняющее вычислимость логарифма.
- Выход. показатель x, для которого $a^x=b(mod p)$, если такой показатель существует.

1. Выбрать произвольные целые числа u, v и положить $c=a^u b^v (mod p), d=c$
2. Выполнять $c=f(c)(mod p), d=f(f(d))(mod p), вычисляя при этом логарифмы для c и d как линейные функции от x по модулю r, до получения равенства $c=d (mod p)$
3. Приняв логарифмы для c и d, вычислить логарифм x решением сравнения по модулю r. Результат x или РЕШЕНИЯ НЕТ.

Оценка сложности

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

Пример работы алгоритма

[26]:
```python
def verify(g, h, p, x):
    # Проверяет заданный набор значений g, h, p и x
    return pow(g, x, p) == h

args = [
    (10, 64, 107),
]
```

[32]:
```python
for arg in args:
    res = pollard(*arg)
    print(arg, ': ', res)
    print("Validates: ", verify(arg[0], arg[1], arg[2], res))
    print()
```

```
(10, 64, 107) :  20
Validates:  True
```

Выводы

Результаты выполнения лабораторной работы

в конце нашего лабораторная работа, я изучил задачу дискретного логарифмирования.