**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский Авиационный Институт»**

**(Национальный Исследовательский Университет)**

**Институт: №8 «Информационные технологии   
и прикладная математика»   
Кафедра: 806 «Вычислительная математика   
и программирование»**

Лабораторная работа № 5   
по курсу «Криптография»

Группа: М8О-306Б-21

Студент(ка): О. А. Мезенин

Преподаватель: А. В. Борисов

Оценка:

Дата: 05.05.2024

Москва, 2024

**ОГЛАВЛЕНИЕ**

[1 Тема 3](#_Toc158983147)

[2 Задание 3](#_Toc158983148)

[3 Теория 4](#_Toc158983149)

[4 Ход лабораторной работы 5](#_Toc158983150)

[5 Выводы 8](#_Toc158983151)

# **Тема**

Эллиптические кривые.

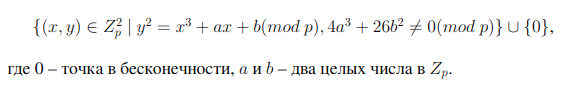
# **Задание**

Подобрать такую эллиптическую кривую, порядок точки которой полным перебором находится за 10 минут на ПК. Упомянуть в отчёте результаты замеров работы программы, характеристики вычислителя. Также указать какие алгоритмы и/или теоремы существуют для облегчения и ускорения решения задачи полного перебора.

Рассмотреть для случая конечного простого поля Z\_p.

# **Теория**

Эллиптическая кривая над конечным простым полем Z\_p (где p — простое число) — это множество точек:



Порядок группы эллиптической кривой — это количество точек на кривой.

Порядок точки P (или порядок подгруппы, порожденной точкой P) эллиптической кривой — это наименьшее положительное число n такое, что nP=0.

Для нахождения такого n методом полного перебора нужно перебирать все k=1,2,3,..., пока не будет выполнено равенство kP=0.

Для ускорения алгоритма можно воспользоваться алгоритмом Шуфа для поиска порядка группы эллиптической кривой и теоремой Лагранжа, согласно которой порядок подгруппы — это делитель порядка исходной группы:

1) Вычисляем порядок эллиптической кривой N с помощью алгоритма Шуфа.

2) Находим все делители N.

3) Для каждого делителя n порядка N вычисляем nP.

4) Наименьшее n, такое, что nP = 0, является порядком подгруппы.

# **Ход лабораторной работы**

Характеристики ПК:  
Процессор AMD Ryzen 5 2600  
ОЗУ 16 ГБ

Реализация класса эллиптической кривой была взята отсюда — <https://github.com/andreacorbellini/ecc/blob/master/logs/common.py> (с небольшими доработками).

Идея алгоритма перебора была в том, чтобы перебирать кривые с разными p (простые числа, найденные с помощью решета Эратосфена), затем находить k первых точек, принадлежащей кривой и для каждой из них уже находить порядок, замеряя время. Так продолжать до тех пор, пока не закончатся простые числа или не будет достигнуто нужное время.

Реализация функции get\_order\_of\_point:

def get\_order\_of\_point(curve, point):  
 p = point  
 order = 0  
 while p := curve.add(p, point):  
 order += 1  
  
 return order

Реализация функции get\_first\_n\_points:

def get\_first\_n\_points(curve, n):  
 points = []  
 for x in range(curve.p):  
 for y in range(curve.p):  
 if curve.is\_on\_curve((x, y)):  
 points.append((x, y))  
 if len(points) >= n:  
 return points

Реализация функции main:

def main():  
 n = int(input())  
 a = int(input())  
 b = int(input())  
 required\_time = int(input()) # in seconds  
 prime\_step = int(input())  
 start\_step = int(input())  
  
 primes = get\_primes(n)  
 max\_time = 0  
 max\_point = 0  
 max\_p = 0  
 max\_point\_order = 0  
 cur\_i = start\_step  
 while max\_time < required\_time and cur\_i < len(primes):  
 p = primes[cur\_i]  
 curve = EllipticCurve(p, a, b)  
 points = get\_first\_n\_points(curve, 2)  
 for point in points:  
 start\_time = time.time()  
 point\_order = get\_order\_of\_point(curve, point)  
 end\_time = time.time()  
 cur\_time = end\_time - start\_time  
 if max\_time < cur\_time:  
 max\_time = cur\_time  
 max\_point = point  
 max\_p = p  
 max\_point\_order = point\_order  
 print(f"p={p}, cur\_time={cur\_time}, point={point}")  
 cur\_i += prime\_step  
  
 print(f"a={a}, b={b}, p={max\_p}, time={max\_time} seconds, point order={max\_point\_order}, point={max\_point}")

Здесь для ускорения поиска были добавлены prime\_step и start\_step — шаг по простым числам и начальный индекс простых чисел. Для каждой кривой берутся две точки.

Было очень много попыток подбора чисел n, prime\_step и start\_step: чаще всего это были недостаточно большие числа, и с такими параметрами пришлось бы ждать целевую кривую очень долго.

Итоговый запуск программы прошел с такими входными данными:

n=400000000  
a=-2  
b=2  
required\_time=600  
prime\_step=100000  
start\_step=5000000

Вывод программы:

p=86028157, cur\_time=128.15531587600708, point=(1, 1)  
p=86028157, cur\_time=129.25277638435364, point=(1, 86028156)  
p=87857533, cur\_time=133.16771912574768, point=(1, 1)  
p=87857533, cur\_time=131.91567468643188, point=(1, 87857532)  
p=89687693, cur\_time=108.58084774017334, point=(1, 1)  
p=89687693, cur\_time=109.79701566696167, point=(1, 89687692)  
p=91519081, cur\_time=280.70820450782776, point=(0, 10706785)  
p=91519081, cur\_time=280.5695593357086, point=(0, 80812296)  
p=93354689, cur\_time=566.002200126648, point=(0, 44947215)  
p=93354689, cur\_time=568.2784616947174, point=(0, 48407474)  
p=95189093, cur\_time=190.49081206321716, point=(1, 1)  
p=95189093, cur\_time=191.45387506484985, point=(1, 95189092)  
p=97026263, cur\_time=293.99523973464966, point=(0, 6576277)  
p=97026263, cur\_time=291.6543004512787, point=(0, 90449986)  
p=98866931, cur\_time=595.0886433124542, point=(1, 1)  
p=98866931, cur\_time=606.4621770381927, point=(1, 98866930)  
a=-2, b=2, p=98866931, time=606.4621770381927 seconds, point order=98866926, point=(1, 98866930)

Найдена кривая с параметрами a=-2, b=2, p=98866931. Порядок точек для неё искался примерно 10 минут (595 и 606 секунд для двух точек).

# **Выводы**

В ходе лабораторной работы познакомился с понятием эллиптических кривых, в том числе над конечным простым полем, и посмотрел, как они применяются в криптографии. Изучил операции, применяемые над эллиптическими кривыми, и такие понятия, как порядок группы и порядок точки. Методом перебора подобрал эллиптическую кривую, порядок точки который находится примерно за 10 минут. Также посмотрел алгоритм, который может ускорить нахождения порядка точки.

# **Список используемой литературы**

1. Доступно о криптографии на эллиптических кривых — <https://habr.com/ru/articles/335906/>