Київський національний університет імені Тараса Шевченка
Факультет комп’ютерних наук та кібернетики
Кафедра інтелектуальних програмних систем
Математичні основи захисту інформації

Лабораторна робота №5
“Імплементація алгоритмів з теорії чисел”

Виконали студенти 3-го курсу
Групи ІПС-32

Роботу виконали:
Ольховатий Ігор
Ковальов Володимир
Тряско Софія
Цілинко Олександр
Бондар Юлія
Волик Артем

Київ 2023

Тема: Імплементація алгоритмів для обчислення великих степенів, розкладання числа на прості множники та знаходження дискретного логарифму.

Мета: Метою цієї лабораторної роботи було імплементувати наступні алгоритми та створити інтерактивний інтерфейс для їх демонстрації:

1. Модулярне піднесення до степені зі швидким підрахунком (метод **modular_exponentiation1**).
2. Модулярне піднесення до степені (метод **modular_exponentiation2**).
3. Розкладання числа на прості множники за допомогою алгоритму Pollard's Rho (метод **rho_algorithm**).
4. Знаходження дискретного логарифму за допомогою алгоритму Pohlig-Hellman (метод **pohlig_hellman**).

Виконання:

1. Розроблено імплементації вказаних алгоритмів на мові програмування Python та включено їх до програми.
2. Створено інтерактивний інтерфейс для використання цих алгоритмів, що дозволяє користувачу обрати алгоритм для виконання.

Використання програми:

Після запуску програми користувач отримує наступне меню:

- 0. Exit
- 1. Модулярне піднесення до степені (метод **modular_exponentiation1**)
- 2. Модулярне піднесення до степені (метод **modular_exponentiation2**)
- 3. Розкладання числа на прості множники (метод **rho_algorithm**)
- 4. Знаходження дискретного логарифму (метод **pohlig_hellman**)

Користувач обирає бажаний алгоритм за допомогою введення відповідного номеру. Після вибору алгоритму введення даних, що необхідні для виконання алгоритму, і програма повертає результат.

Зразок роботи програми:

1. Використання модулярного піднесення до степені (метод **modular_exponentiation1**):

Enter the algorithm name: 1

This algorithm computes the a^d (mod m)

Enter values for a, d, m

(Вхідні дані)

a = 5

d = 3

m = 7

(Вихід програми)

5^3 (mod 7) = 6

2. Використання модулярного піднесення до степені (метод **modular_exponentiation2**):

Enter the algorithm name: 2
This algorithm computes the a^d (mod m)
Enter values for a, d, m

(Вхідні дані)

a = 2
d = 1000
m = 1001

(Вихід програми)

2^1000 (mod 1001) = 562

3. Використання алгоритму Pollard's Rho (метод **rho_algorithm**):

Enter the algorithm name: 3

Pollard's Rho Algorithm Demo

(Вхідні дані)

Enter an integer greater than 2 (or enter 0 to quit): 221

(Вихід програми)

A non-trivial divisor of 221 is 13

Тепер тест для простого числа:
Enter the algorithm name: 3

Pollard's Rho Algorithm Demo

(Вхідні дані)

Enter an integer greater than 2 (or enter 0 to quit): 104723

(Вихід програми)

No non-trivial divisor found for 104723

4. Використання алгоритму Pohlig-Hellman (метод **pohlig_hellman**):
   Enter the algorithm name: 4

   Pohling-Hellman's algorithm for discrete logarithm

   Formula: h ≡ g^x (mod p)

   (Вхідні дані)

   h: 525
   g: -2
   p: 131

   (Вихід програми)

   Solving 525 ≡ -2^x (mod 131)
   ---
   q | e | g^((p-1)/q^e) | h^((p-1)/q^e) | Solve (g^((p-1)/q^e))^x = h^((p-1)/q^e) for x
   ---
   2 | 1 | 1 | 1 | x ≡ 0 (mod 2)
   ---
   13 | 1 | 107 | 1 | x ≡ 0 (mod 13)

5	1	53	1	x ≡ 0 (mod 5)

Solution x = 0

Висновок:
В результаті цієї лабораторної роботи було успішно імплементовано алгоритми для обчислення великих степенів, розкладання числа на прості множники та знаходження дискретного логарифму. Також, був створений інтерактивний інтерфейс, що дозволяє користувачам легко використовувати ці алгоритми.

Література:

- Лекції з предмету “Математичні основи захисту інформації”
- https://home.cs.colorado.edu/~srirams/courses/csci2824-spr14/pollardsRho.html
- https://en.wikipedia.org/wiki/Pohlig%E2%80%93Hellman_algorithm
- https://stackoverflow.com/questions/57668289/implement-the-function-fast-modular-exponentiation