**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**

**«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ СІКОРСЬКОГО»**

**Інститут прикладного системного аналізу**

**Кафедра системного проектування**

Лабораторна робота №3

з дисципліни

“Безпека інформаційних систем”

Виконав:

студент групи ДА-81

Дєрюгін Єгор Олександрович

# Київ – 2021**Мета роботи**

Дослідження криптосистеми шифрування даних RSA за допомогою програмних засобів CryptTool та методики її практичної реалізації.

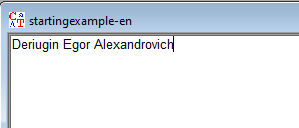
**Завдання**

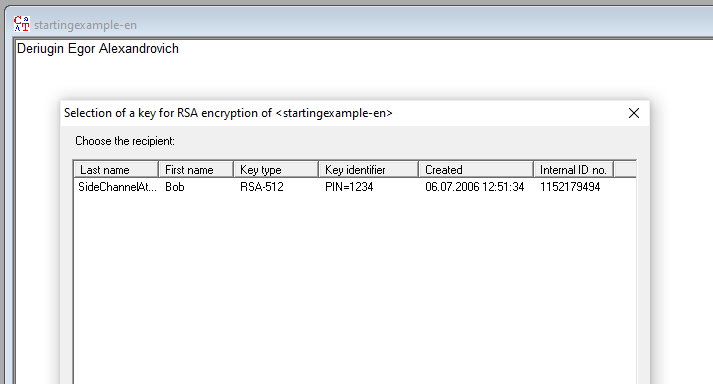
1. Зашифрувати вручну своє прізвище, ім’я та по-батькові, написані великими та малими буквами англійського алфавіту з урахуванням пробілів, за допомогою шифру RSA. Для цього за допомогою пакету CrypTool вибрати параметри криптосистеми RSA з розрахунком, щоб можна було виконати шифрування і дешифрування вручну за допомогою інженерного калькулятора. При кодуванні букв використовувати кодову таблицю ASCII, параметри p і q криптосистеми RSA вибрати в межах від 27 до 28 . Виконати вручну дешифрування отриманого шифротексту.
2. Виконати шифрування і дешифрування свого прізвища, імені та по-батькові за допомогою засобів пакету CrypTool. Зберегти отримані результати шифрування і дешифрування у відповідні файли за допомогою засобів пакету CrypTool. Порівняти результати ручного та автоматичного шифрування і дешифрування.
3. Вибрати варіант текстового файлу (див. Додаток до лабораторної роботи № 1) відповідно до порядкового номера студенту в списку академічної групи. За допомогою засобів пакету CrypTool зашифрувати і дешифрувати обраний файл шифром RSA з параметрами, які використовувалися при виконанні п.2. Порівняти вихідний текст з дешифрованим. Зробити висновки. Результати зберегти.
4. За допомогою засобів аналізу пакету CrypTool провести частотний аналіз вихідного і зашифрованого тексту. Визначити ентропію і максимально можливу ентропію зашифрованого файлу, а також побудувати гістограму розподілу частоти символів в аналізованих файлах. Зробити висновки про можливість криптоаналізу зашифрованого тексту з використанням тільки шифротексту. Дослідити можливості пакета CrypTool для злому шифру RSA.
5. Написати і налагодити програму для шифрування і дешифрування файлів за допомогою шифру RSA використовуючи готові бібліотеки (наприклад  System.Security.Cryptography в середовищі .NET Security Framework). Зашифрувати текстовий файл, вибраний з додатку до лабораторної роботи при виконанні п.3, за допомогою розробленої програми. Порівняти результати шифрування, отримані за допомогою пакета CrypTool, з результатами роботи власної програми. Зробити висновки про коректність роботи програми.
6. Оформити звіт по роботі.

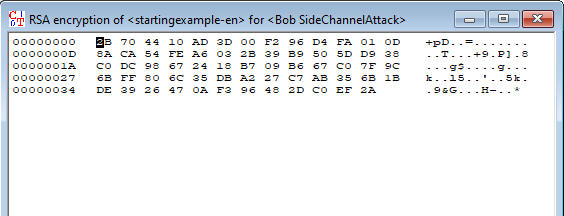
Хід роботи

2. Виконуємо шифрування і дешифрування прізвища, імені та по-батькові за допомогою CrypTool.

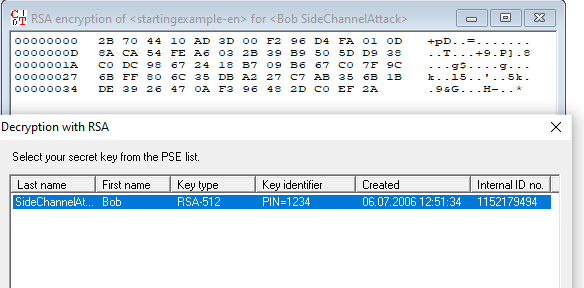
Шифрування:

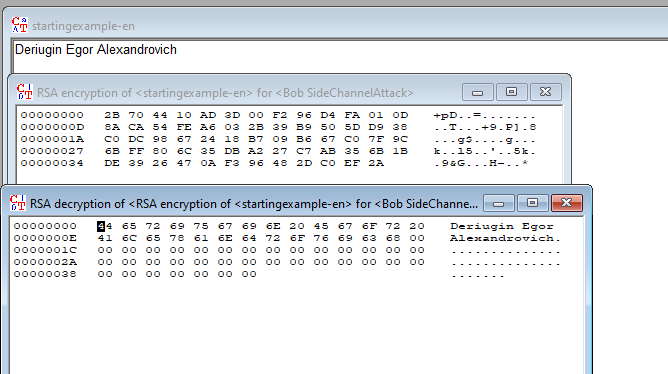






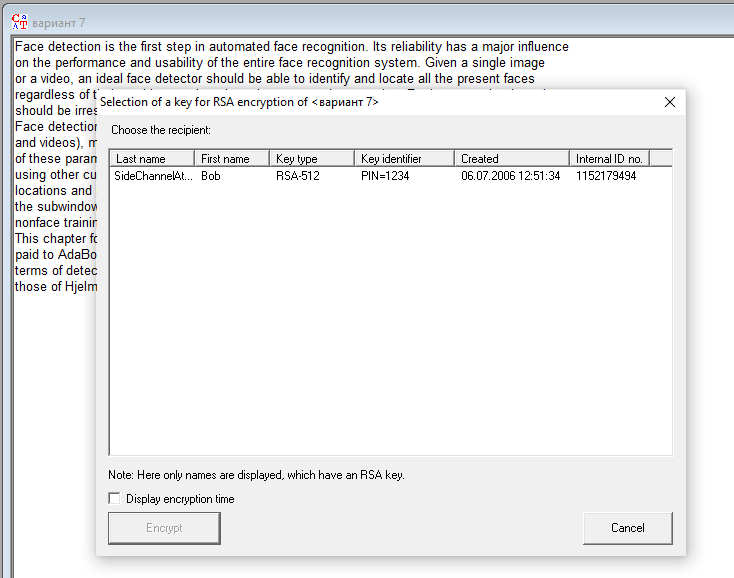
Дешифрування:

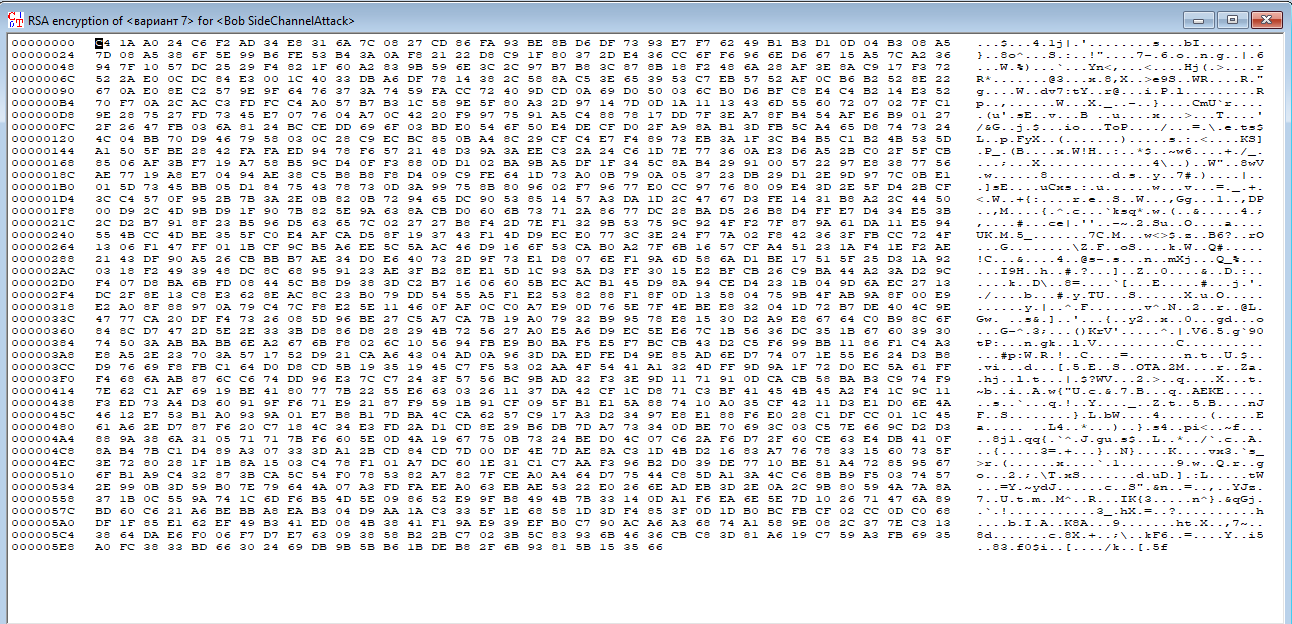




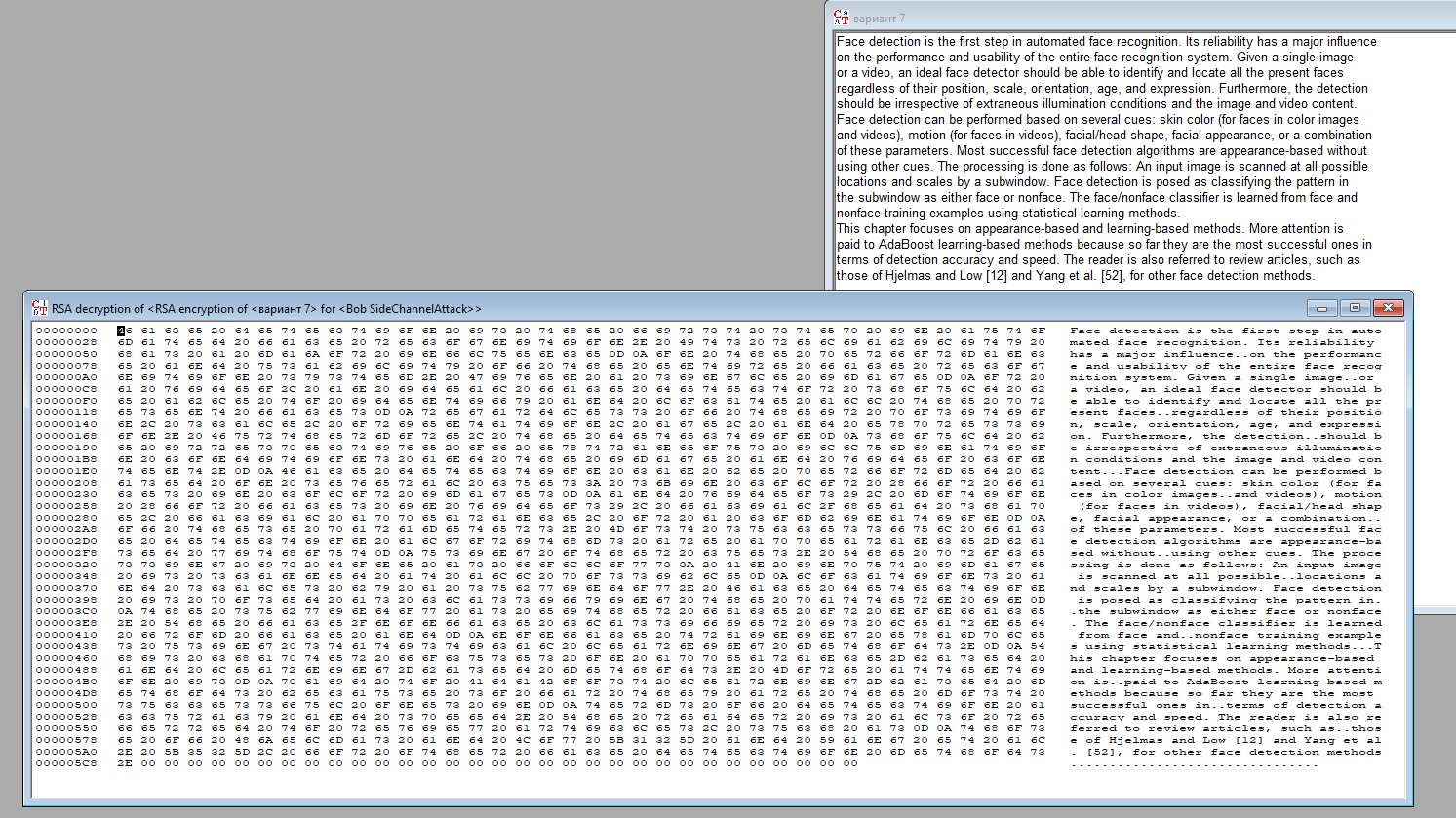
3. Зашифрувати і дешифрувати текст з файлу, згідно варіанту, використовуючи CrypTool.

Шифрування:



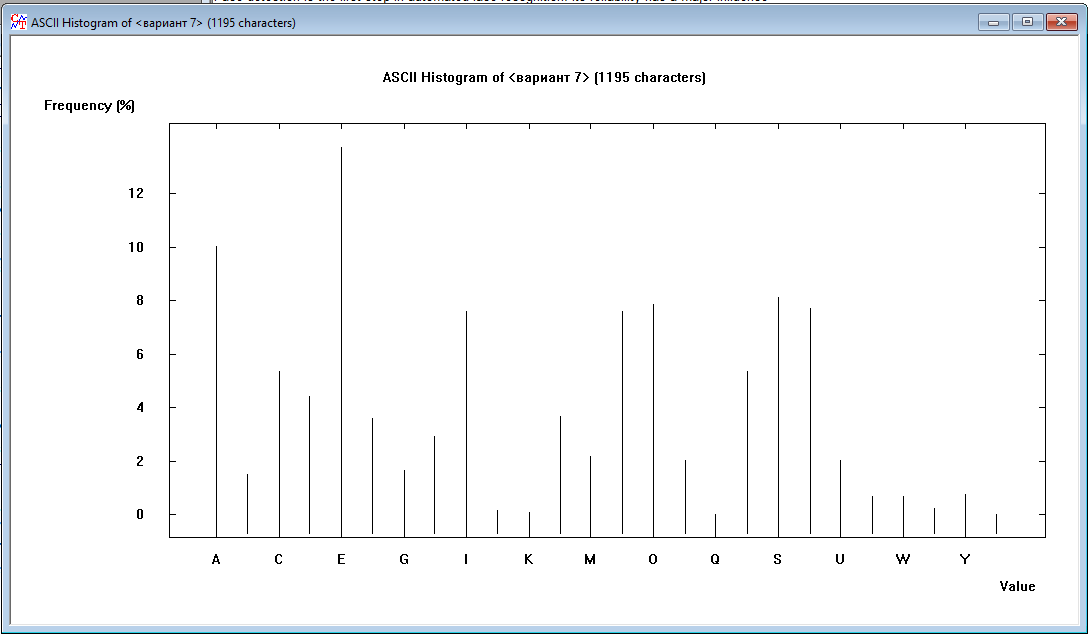


Дешифрування:

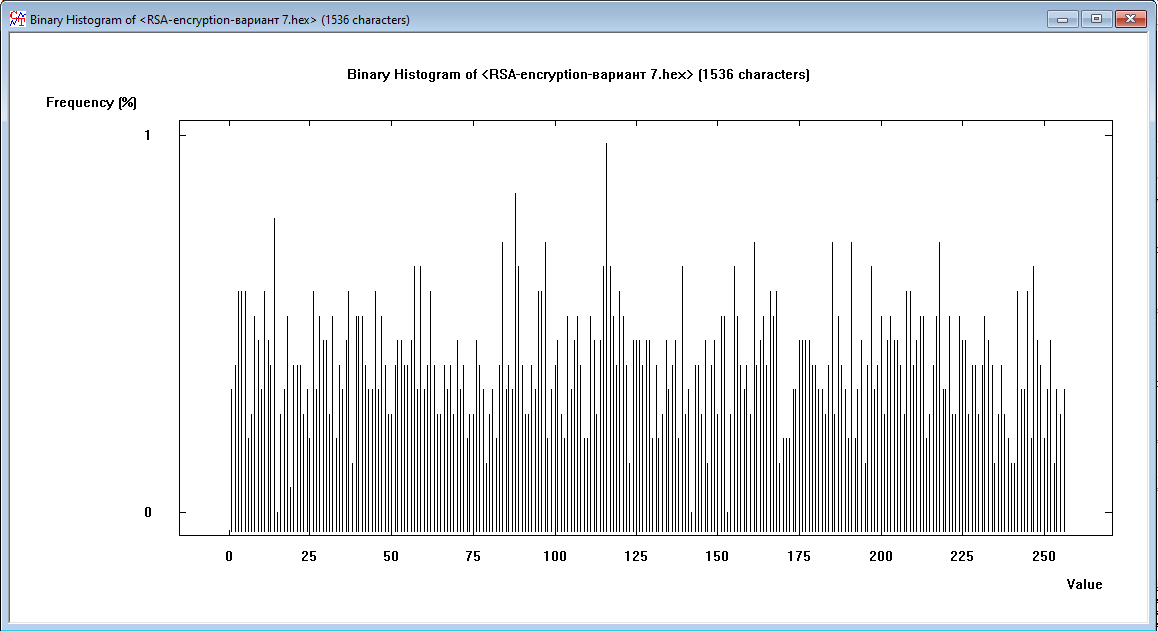


4. За допомогою засобів аналізу пакету CrypTool провести частотний аналіз вихідного і зашифрованого тексту. Визначити ентропію і максимально можливу ентропію зашифрованого файлу, а також побудувати гістограму розподілу частоти символів в аналізованих файлах.

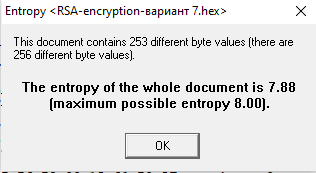
Гістограма розподілу частоти символів початкового тексту:



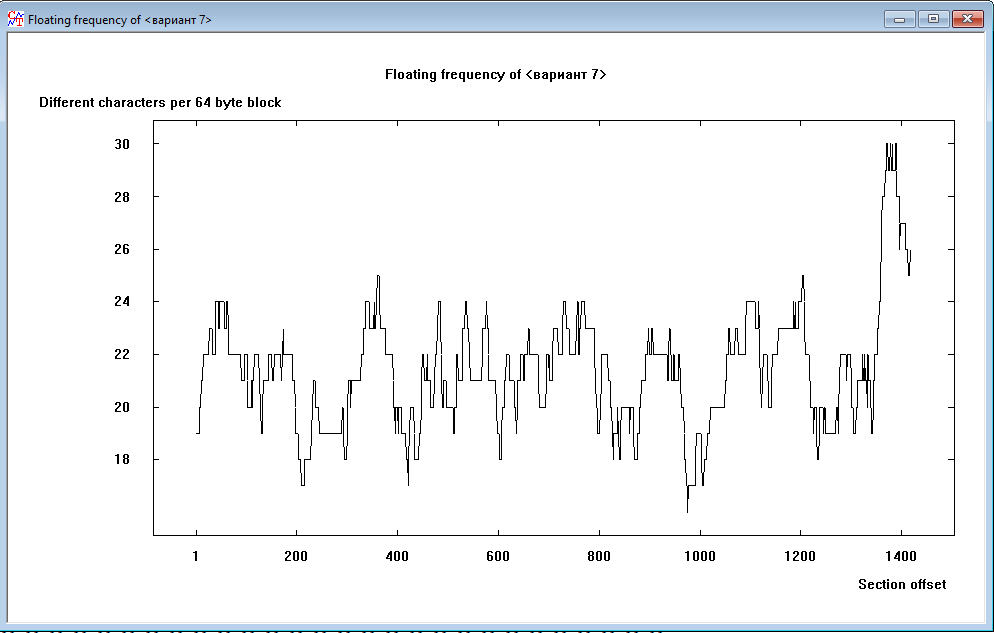
Гістограма розподілу частоти символів зашифрованого тексту:



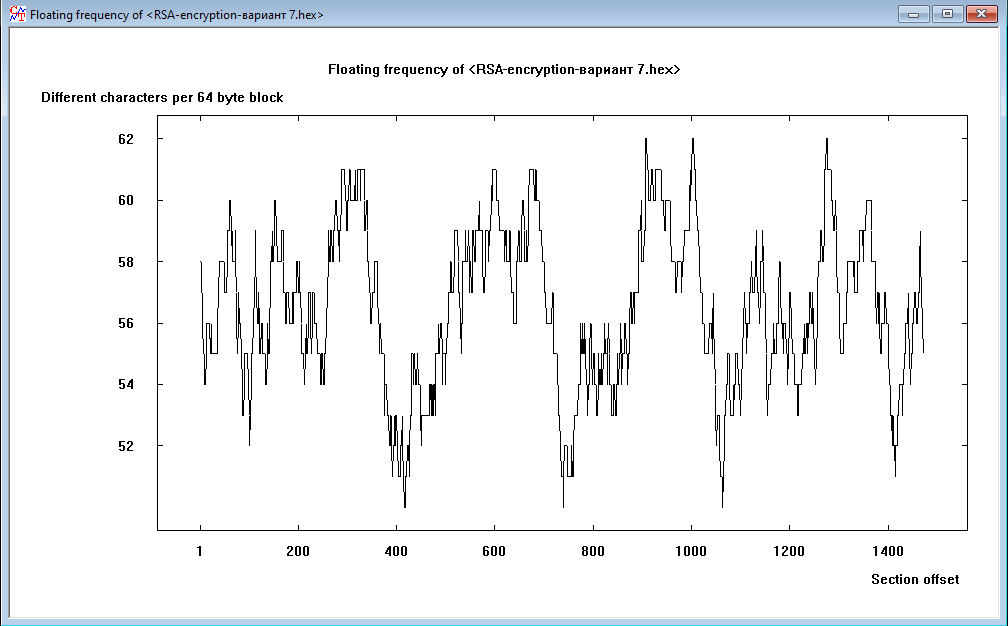
Ентропія та максимально можлива ентропія зашифрованого тексту:



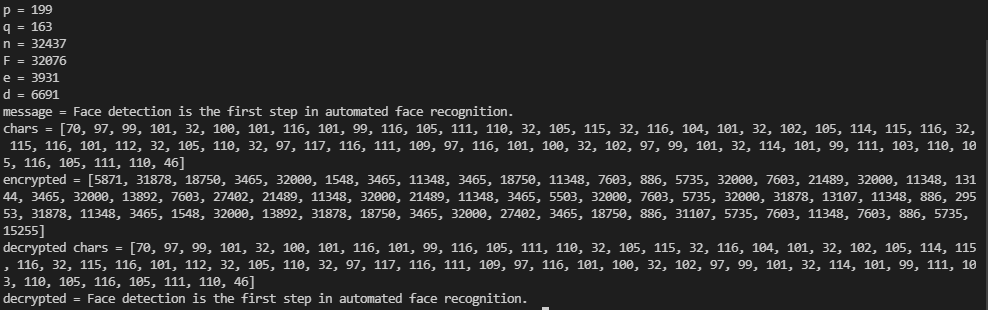
Частотний аналіз початкового тексту:



Частотний аналіз зашифрованого тексту:



5. Результат роботи програми:



Лістинг програми:

import random

def isPrime(n):

    if n & 1 == 0:

        return False

    d = 3

    while d \* d <= n:

        if n % d == 0:

            return False

        d = d + 2

    return True

primes = [i for i in range(2 \*\* 7, 2 \*\* 8) if isPrime(i)]

p = 199

primes.remove(p)

q = 163

n = p \* q

F = (p - 1) \* (q - 1)

e = 3931

s = isPrime(140)

get\_d = lambda x: ((F \* x) + 1) / e

i = 1

d = get\_d(i)

while d % 1 != 0:

    i += 1

    d = get\_d(i)

d = int(d)

print(f"p = {p}\n"

      f"q = {q}\n"

      f"n = {n}\n"

      f"F = {F}\n"

      f"e = {e}\n"

      f"d = {d}")

message = "Face detection is the first step in automated face recognition."

chars = list(map(lambda x: ord(x), message))

encrypted = list(map(lambda x: x \*\* e % n, chars))

dec\_chars = list(map(lambda x: x \*\* d % n, encrypted))

decrypted = list(map(lambda x: chr(x), dec\_chars))

print(f"message = {message}\n"

      f"chars = {chars}\n"

      f"encrypted = {encrypted}\n"

      f"decrypted chars = {dec\_chars}\n"

      f"decrypted = {''.join(decrypted)}")

Висновки:

У ході виконання лабораторної роботи було вивчено асиметричне шифрування RSA. Виконано шифрування/дешифрування тексту за допомогою CrypTool, а також проведено аналіз початкового та зашифрованого текстів.

Було написано програмний код, який реалізує шифрування та дешифрування RSA.