Государственное учреждение образования

“БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ”

Кафедра: Интеллектуальных информационных технологий

Дисциплина: Средства и методы защиты информации в интеллектуальных системах

**Отчет по лабораторной работе №5**

**“Ассиметричное шифрование и электронная цифровая подпись”**

Выполнил:

студент гр.221701

Телица И.Д.

Проверил:

Захаров В.В.

Минск 2024

#### **Задача:**

Разработать программное обеспечение, реализующее функции генерации секретного и открытого ключей, шифрования и цифровой подписи для алгоритма RSA. Обмен входными и выходными данными должен осуществляться через файлы:

– открытого ключа;

– секретного ключа;

– исходного сообщения;

– зашифрованного сообщения.

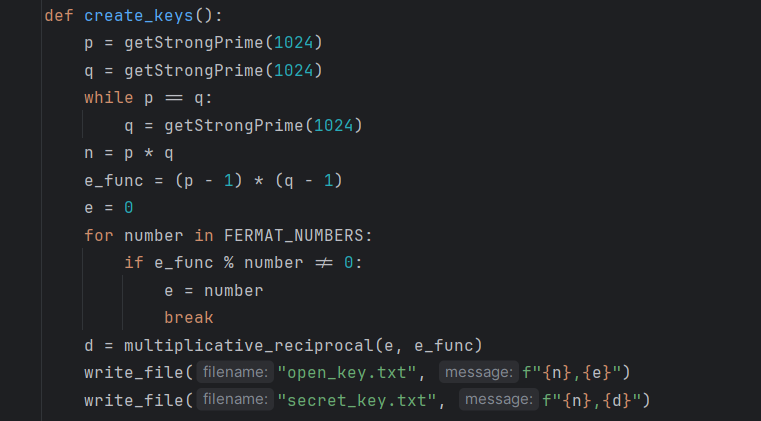
Для повышения скорости шифрования использовать метод последовательного возведения в квадрат и умножения. Длина чисел p и q должна быть не менее 1024 бит.

## **Ход работы:**

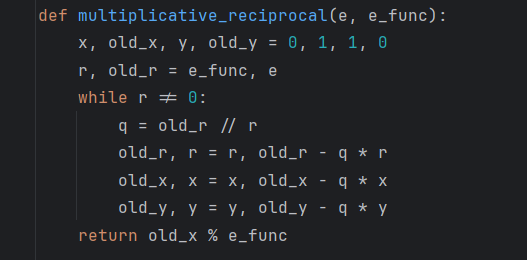
1. Создание ключей:

Функция **create\_key()** использует функцию **getStrongPrime()** модуля **Crypto.Util.number** для получения больших простых чисел.

Вычисляется значения n = p∙q и функции Эйлера

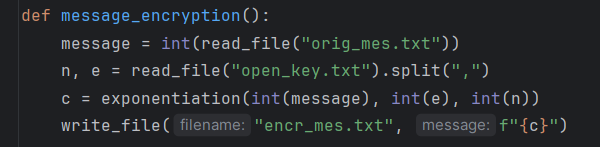


Для числа е из , которое является взаимно простым с , с помощью функции **multiplicative\_reciprocal()** находим число d, мультипликативно обратное к числу e по модулю n, удовлетворяющее сравнению: . Пары n, e и n, d записываются в файлы в качестве открытого и закрытого ключей соответственно.

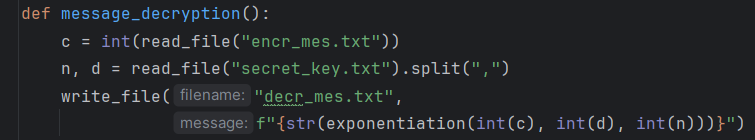


Функция **multiplicative\_reciprocal()** использует расширенный алгоритм Евклида для нахождения НОД и коэффициентов соотношения Безу.

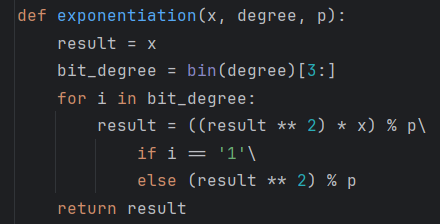
1. Шифрование и дешифрование сообщений:

В качестве исходного сообщения m выступают целые числа в интервале от 0 до n. Для зашифрования используют открытый ключ и вычисляют криптограмму 

Для расшифрования используют секретный ключ и вычисляют исходное сообщение

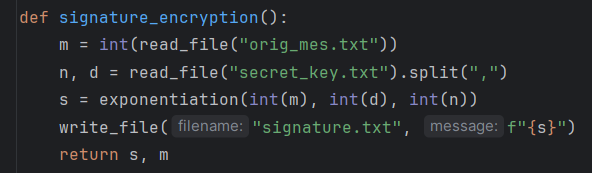


Метод **exponentiation(x, degree, p)** производит возведение в степень методом последовательного возведения в квадрат и умножения, операция взятия по модулю производится на каждом шаге.

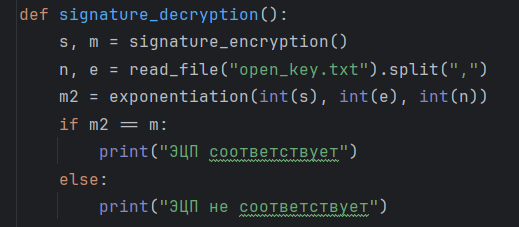


1. Создание и проверка ЭЦП:

Для создания цифровой подписи s с помощью секретного ключа вычисляют

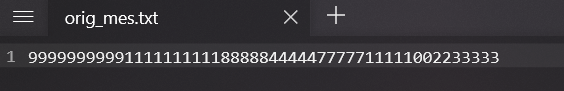


Проверяющий вычисляет прообраз сообщения из подписи m\* = , сравнивает m и m\*. Если они равны, значит подпись верна, в противном случае – ложна.

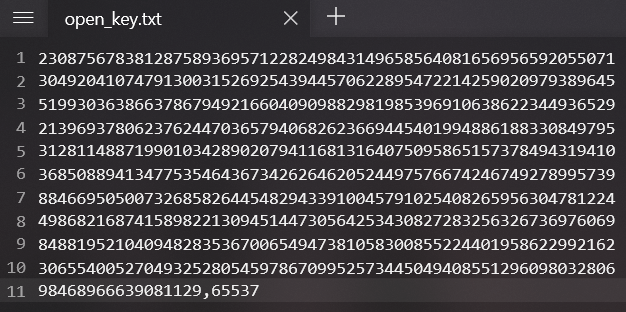


## **Результат работы:**

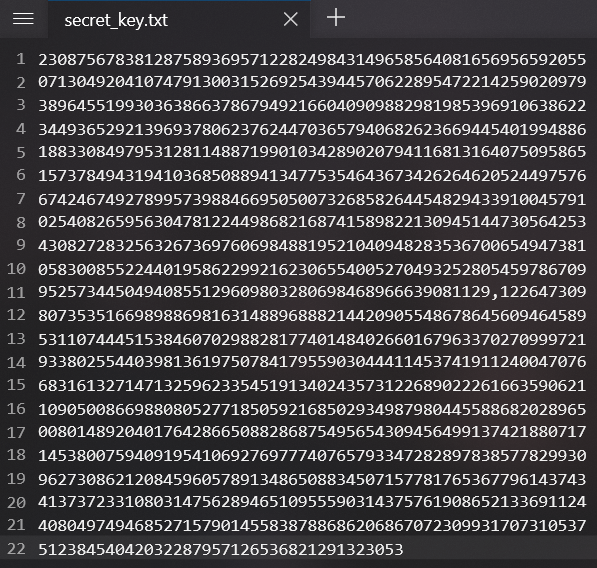
Исходное сообщение:



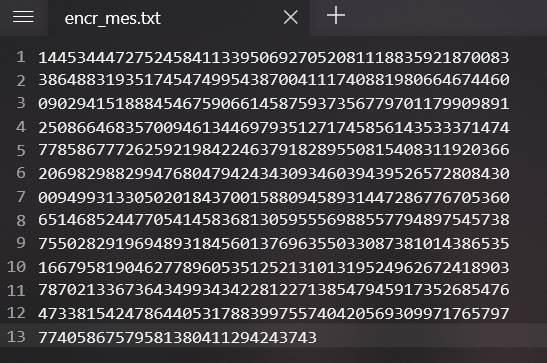
Сгенерированный открытый ключ:



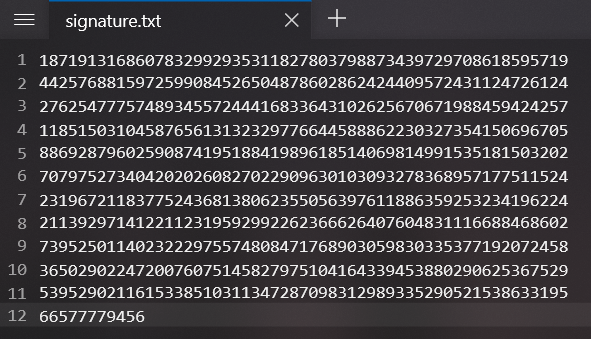
Сгенерированный секретный ключ:



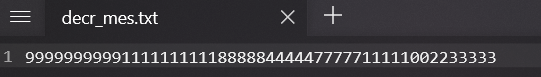
Зашифрованное сообщение:



Сгенерированная ЭЦП:



Расшифровка сообщения и проверка ЭЦП:





#### **Вывод**:

В ходе лабораторной работы были изучены, реализованы и протестированы программные средства шифрования и дешифрования при помощи алгоритма RSA, использующего принцип асимметричного шифрования с помощью открытых и секретных ключей.