**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики и информатики**

**Отчет**

**Лабораторная работа № 5**

**Ассиметричная криптография**

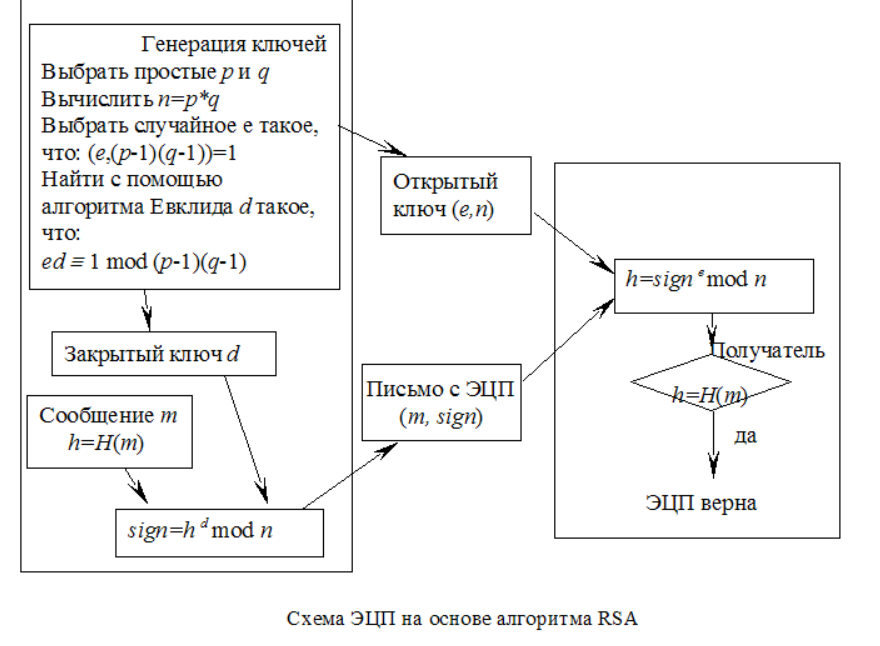
Выполнил студент группы №12

*Шишлянников Иван Викторович*

**Минск 2020**

# Реализация цифровой подписи

Для реализации реализации ЭЦП RSA я пользовался следующей схемой:

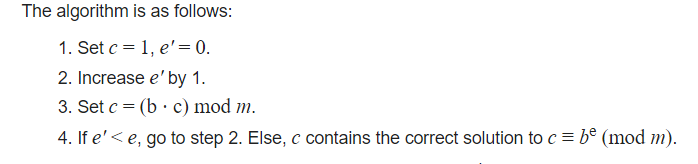


Программа генерирует случайные p и q (простые), случайное число e, соответствующее условию (на одну и ту же пару p и q могут генерироваться различные значения e). Закрытый ключ d находим при помощи расширенного алгоритма Евклида.

Далее сгенерированные значения сохраняются в файл для дальнейшего использования.

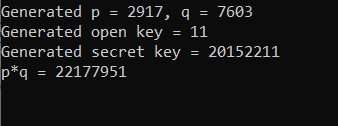
В качестве Хэш – функции, я использовал блочно – итерационный алгоритм, взятый из лабораторной работы 4. Данный алгоритм основан на DES. После выполнения преобразований (кодирования) сообщения m, я перевожу полученный шифр в число (Хэширую), для этого я просто суммирую все символы в шифре ( по их коду соответственно) и беру значение по модулю p\*q ( чтобы не получить недопустимые значения h так как далее вся работа будет вестись непосредственно с делением по модулю p\*q). Ключ для DES – шифрования я генерирую исходя из открытого ключа e, так как доступ к Хэш – функции есть на обоих концах.

Для избежания переполнений во время возведения в степень по модулю я использовал следующий алгоритм:

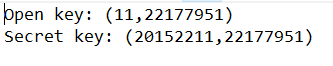


Таким образом полученный h я на стороне получателя сверяю с полученным h исходя из полученной подписи.

Пример работы:



Ключи в .txt файле:



Теперь попробуем отправить с их помощью сообщение с подписью и проверить его на стороне получателя на валидность.

Отправим следующее сообщение:



Полученное значение хэша нашего сообщения:



Созданная подпись на стороне отправителя:



Проверка подписи на валидность:



## Примечания

Для валидной работы DES входные данные (ключ и текст для шифрования)

Редактируются таким образом, чтобы соответствовать требуемым размерам для шифрования. Для этого я просто дописываю необходимое количество символов справа как для ключа, так и для текста.

Программа может работать и с более крупными значениями, но в целях экономии времени я всю работу проделывал с числами меньше 10^10.