Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

**Исследование асимметричных шифров**

**RSA и Эль-Гамаля**

Студент: Каспер Н.В.

ФИТ 3 курс 5 группа

Преподаватель: Берников В.О.

Минск 2020

1. **Описание приложения**

Приложение написано на языке программирования C# и позволяет:

* зашифровывать сообщение с помощью алгоритма RSA;
* расшифровывать сообщение с помощью алгоритма RSA;
* зашифровывать сообщение с помощью алгоритма Эль-Гамаля;
* расшифровывать сообщение с помощью алгоритма Эль-Гамаля;
* оценивать время выполнения зашифрования и расшифрования.

1. **Методика выполнения поставленных задач**

RSA – первый полноценный алгоритм с открытым ключом, который стал одним из основных алгоритмов для шифрования и для электронно цифровой подписи.

Из всех алгоритмов с открытым ключом – самый простой в реализации и понимании. Безопасность алгоритма обеспечивается за счет трудности разложения на множители больших чисел. Открытый и закрытый ключ – функции двух больших простых чисел. Предполагается, что восстановление открытого текста по шифртексту и открытому ключу эквивалентно разложению на множители двух больших чисел.

Результат работы приложения для алгоритма RSA представлен на рисунке 2.1.

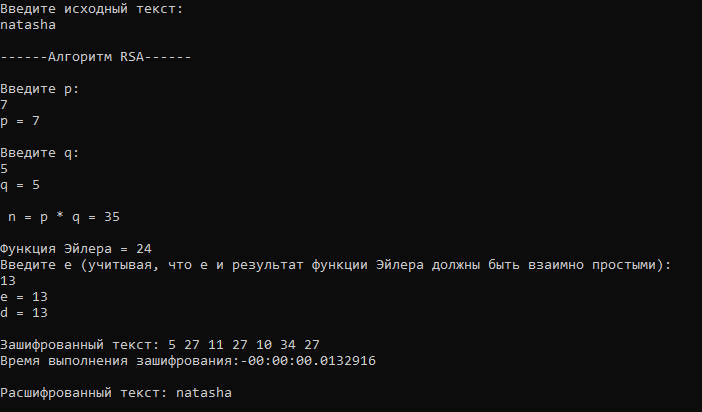


Рисунок 2.1 – Результат работы приложения для RSA

Для генерации тайного и открытого ключа используется 2 больших простых числа p и q. Для большей криптостойкости p и q выбирают равно длины. Дальше рассчитывается n = p\*q – один из 3 компонент ключа.

На рисунке 2.2 представлен программный код, в котором вводится с клавиатуры некоторое простое число p. Также существует проверка введенного числа и если введенное число не является простым – необходимо вводить числа до тех пор, пока введенное число не будет простым

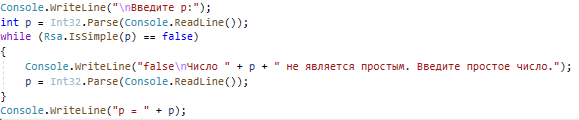


Рисунок 2.2 – Ввод простых чисел

Далее вводится некоторое число e, которое должно быть взаимно простым с (p-1)\*(q-1). Затем вычисляется число d с помощью расширенного алгоритма Евклида, реализация которого представлена на рисунке 2.3.



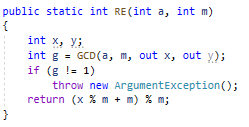


Рисунок 2.3 – Генерация числа d

Следующий шаг – зашифрование сообщения. Зашифрование происходит с помощью ключей (e, n). Также, важно заметить, что длина шифртекста будет равна длине исходного текста. Функция для зашифрования представлена на рисунке 2.4.

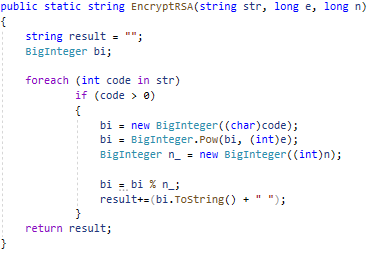


Рисунок 2.4 – Функция для зашифрования сообщения

Для расшифрования сообщения используется пара (d, n). Функция для расшифрования представлена на рисунке 2.5.

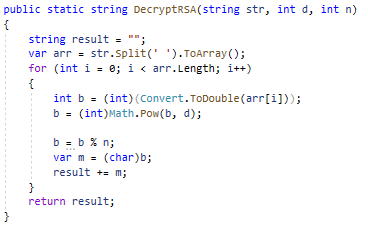


Рисунок 2.5 – Функция для расшифрования сообщения

Результат работы приложения для алгоритма Эль-Гамаля представлен на рисунке 2.6. Безопасность данного алгоритма обоснована трудностями вычисления дискретного логарифма.

В отличии от RSA в данном алгоритме один блок исходного текста заменяется двумя блоками шифртекста. А также при зашифровании используется число k, которое почти никак не связано с ключевой информацией и принимает различные значения при зашифровании и расшифровании.

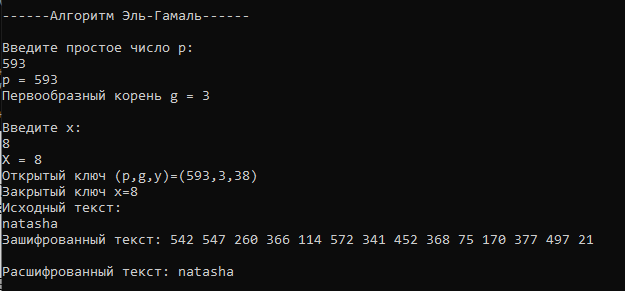


Рисунок 2.6 – Результат работы приложения для Эль-Гамаля

Первоначально выбирается простое число p. Затем выбирается число g (g<p), которое является первообразным корнем по модулю p. Далее выбирается число x (x<p). И на основании данных чисел вычисляется последний компонент y путем возвещения числа g в степень x и деления по модулю p.

Функция для нахождения первообразного корня представлена на рисунке 2.7.

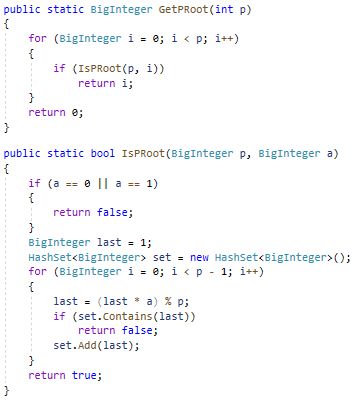


Рисунок 2.7 – Функция нахождения первообразного корня

При зашифровании используются числа p, g, y. Функция зашифрования представлена на рисунке 2.8.

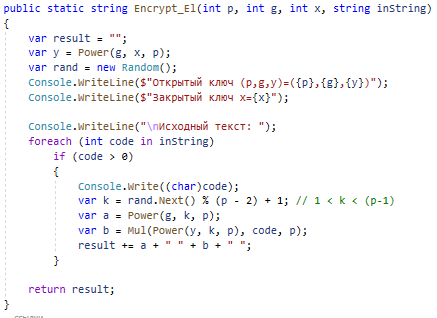


Рисунок 2.8 – Функция зашифрования

При расшифровании используются числа p, g, x. При чем x хранится в тайне. Функция расшифрования представлена на рисунке 2.9.

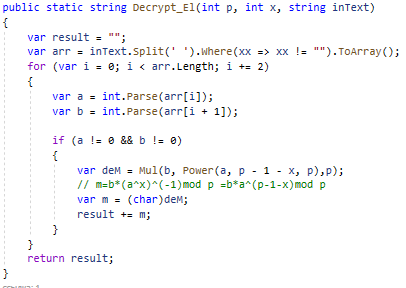


Рисунок 2.9 – Функция расшифрования

Таким образом, были реализованы все поставленные задачи. Были исследованы асимметричные шифры.

**Вывод**

В ходе лабораторной работы были приобретены навыки шифрования используя алгоритм RSA и Эль-Гамаля. Были изучены основные принципы работы асимметричных шифров.

Также было разработано приложение, на языке программирования C#, для реализации задач, связанных с шифрованием данных с помощью алгоритма укладки ранца.