МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образование «Белорусский государственный технологический университет»

Кафедра информационных систем и технологий

**«Исследование асимметричных шифров RSA и Эль-Гамаля»**

Студент:

Септилко Анастасия Антоновна

Преподаватель:

Блинова Евгения Александровна

Минск 2020

**ЗАДАНИЯ:**

**1.** С помощью простого консольного приложения составить табличную или графическую форму зависимости времени вычисления параметра у, функционально заданного выражением вида:

у = ax mod n, от параметров: а (десятичные числа от 5 до 35; можно взять 1 или 2 числа), х (числа, желательно – простые, из диапазона от 103 до 10100; для примера взять 5-10 чисел, равномерно распределенных в указанном диапазоне), n (для примера взять числа, в двоичном виде состоящие из 1024 и 2048 бит).

**2.** Разработать авторское оконное приложение в соответствии с целью лабораторной работы. При этом можно воспользоваться доступными библиотеками либо программными кодами.

В основе вычислений – кодировочные таблицы Base64 и ASCII. Приложение должно реализовывать следующие операции:

• зашифрование и расшифрование текстовых документов на основе алгоритмов RSA и Эль-Гамаля;

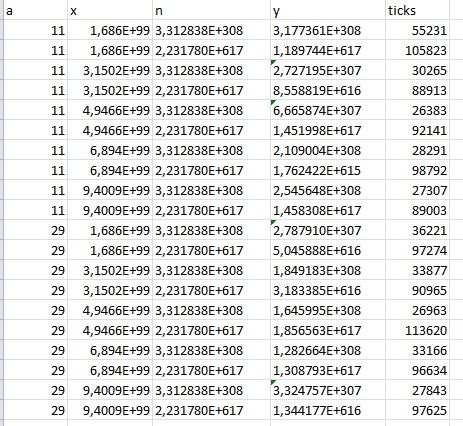
• определение времени выполнения операций.

Исходный текст для зашифрования – собственные фамилия, имя, отчество. Для численного представления блоков текста можно, в том числе, пользоваться указанными выше кодировочными таблицами.

Ключевую информацию для обоих алгоритмов можно сгенерировать самостоятельно либо воспользоваться, например, одной из утилит криптографической библиотеки OpenSSL, с помощью которой, в частности, можно сгенерировать ключевую информацию для алгоритма RSA.

**3.** Используя примерно одинаковый порядок ключевой информации, оценить производительность обоих алгоритмов и относительное изменение объемов криптотекстов (по отношению к объемам открытых текстов).

1)



Для генерации двух ключей: тайного и открытого(а по сути – двух взаимосвязанных частей одного ключа, т. е. ключа, принадлежащего одному физическому лицу (или группе лиц), либо одному юридическому лицу) используются два больших случайных простых числа, p и q. Для максимальной большей криптостойкости нужно выбирать p и q равной длины. Рассчитывается произведение: n = pq. Этой есть один из трех компонент ключа, состоящего из чисел n, e, d. Затем случайным образом выбирается второй компонент ключа(открытый ключ или ключ зашифрования, e, такой что e и (p-1)(q-1) являются взаимно простыми числами; вспомним, что (p-1)(q-1) = φ(n) – функция Эйлера. Б. Шнайер [4] рекомендует число е выбирать из ряда: 3, 17, 216 + 1. Наконец расширенный алгоритм Евклида используется для вычисления третьего компонента ключа: ключа расшифрования, d, такого, что выполняется условие:

ed = 1 (mod φ(n)).

Другими словами: d-1 = e(mod φ(n)).

Таким образом, сформирован ключ, состоящий из трех чисел, которые, в свою очередь, образуют две вышеупомянутые взаимосвязанные части: открытый (публичный) ключ, (e, n), и тайный ключ, (d, n; на самом деле, как видим, тайным здесь является лишь первое из пары чисел).

Функция для зашифования алгоритмом RSA показана на рисунке 1.1. Здесь мы получаем четыре параметра: строку введенную, ключ е (функция Эйлера), число n (произведение просты чисел p и q), алфавит.

**Зашифрование.** Если шифруется сообщение М, состоящее из r блоков: m1, m2 , …, mi,…,mr, то шифртекст С будет состоять из такого же числа (r) блоков, представляемых числами: ci = (mi)e mod n.

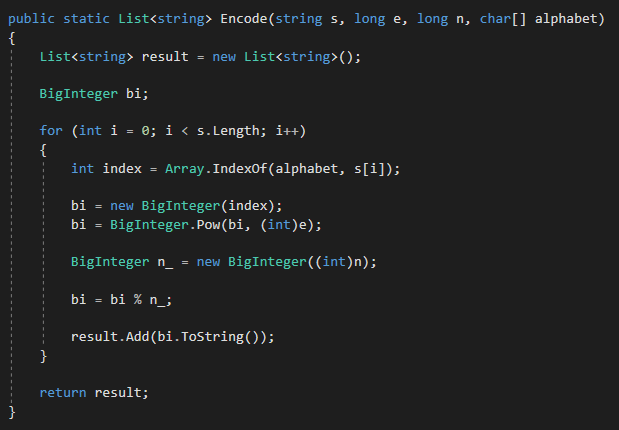


Рисунок 1.1 – Функция шифрования

На рисунке 1.2 приведена функция для расшифрования по алгоритму RSA.

**Расшифрование.** Для расшифрования каждого зашифрованного блока производится вычисление вида:

mi = (ci)d mod n.

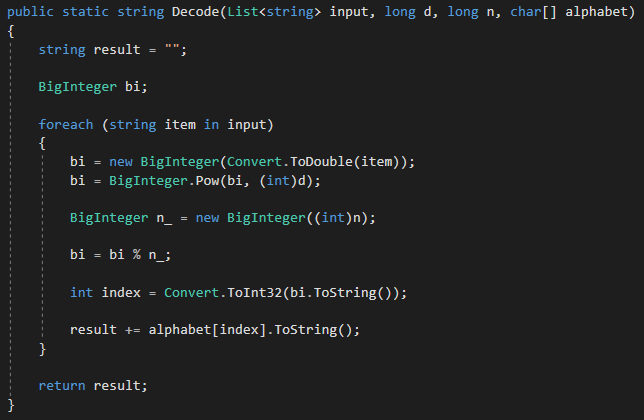


Рисунок 1.2 – Функция расшифрования

Пример выполнения этих функций приведены на рисунке 1.3

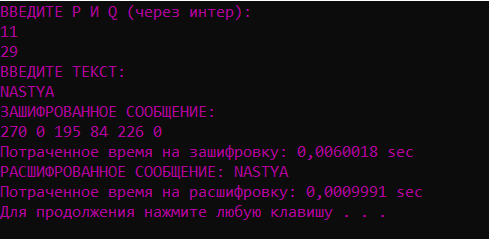


Рисунок 1.3 – Результат выполнения алгоритма RSA

Алгоритм Эль-Гамаля отличается от алгоритма RSA несколькими параметрами и особенностями:

1) генерацией ключевой информации и числом компонент, составляющих ключ;

2) каждому блоку (символу) открытого сообщения в шифртексте на основе алгоритма Эль-Гамаля соответствуют 2 блока (в RSA – один-один);

3) в алгоритме Эль-Гамаля при зашифровании используется число (обозначим его k), которое практически никак не связано с ключевой информацией получателя и которое принимает (по определению) различные значения при зашифровании различных блоков сообщения.

Для работы с алгоритмом была разработана функция Power (рисунок 1.4). Здесь эта функция выполняет роль возведение в степень чисел, а и b. Данная функция часто используется приложением.

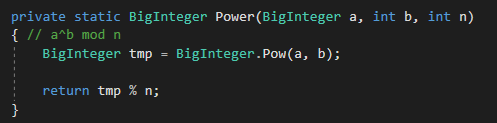


Рисунок 1.4 – Функция Power()

Функция Crypt (рисунок 1.5). Эта функция выполняет шифрование сообщений.

**Зашифрование сообщения.** Как ранее, предположим, что сообщение М = {mi}, где – mi – i-й блок сообщения. Зашифрование отправителем (каждого отдельного блока mi исходного сообщения) предусматривает использование, как это особо подчеркивалось выше, некоторого случайного числа k (1 < k <p – 1). В силу использования случайной величины k шифр Эль-Гамаля называют также шифром многозначной замены, а также схемой вероятностного шифрования. Вероятностный характер шифрования является преимуществом для схемы Эль-Гамаля по сравнению, например, с алгоритмом RSA. Блок шифртекста (ci) состоит из двух чисел: аi и bi:

ai = gk mod p, bi = (yk \*mi) mod p.

Здесь стал очевидный упомянутый недостатком алгоритма шифрования Эль-Гамаля: удвоение (реально – примерно в 1,5 раза) длины зашифрованного текста по сравнению с начальным текстом.

Случайное число k должно сразу после вычисления уничтожаться.

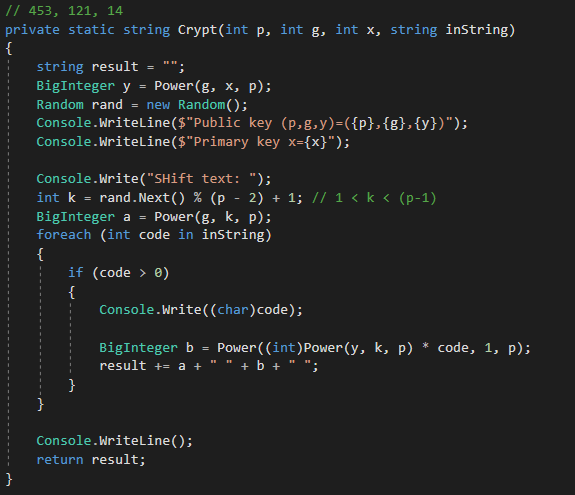


Рисунок 1.5 – Функция Crypt()

Функция для расшифрования приведена ниже (рисунок 1.6).

**Расшифрование сообщения.** ci выполняется по следующей формуле:

mi = (bi \*(ai)x)-1) mod p

или

mi = (bi \*(ai)р-x-1) mod p

где (ax)-1 – обратное значение числа ax по модулю p. Нетрудно проверить, что (ai)x)-1) = gkх mod p.

Еще раз возвратимся к криптостойкости рассмотренного алгоритма. Если для зашифрования двух разных блоков (m1и m2) некоторого сообщения использовать одинаковые k, то для соответствующих шифртекстов c1 = (a1, b1) и c2 = (a2, b2) выполняется соотношение b1(b2)-1 = m1(m2)-1. Из этого выражения можно легко вычислить m2, если известно m1.

При примерно одинаковой размерности ключей рассмотренные алгоритмы обеспечивают примерно одинаковый уровень криптостойкости.

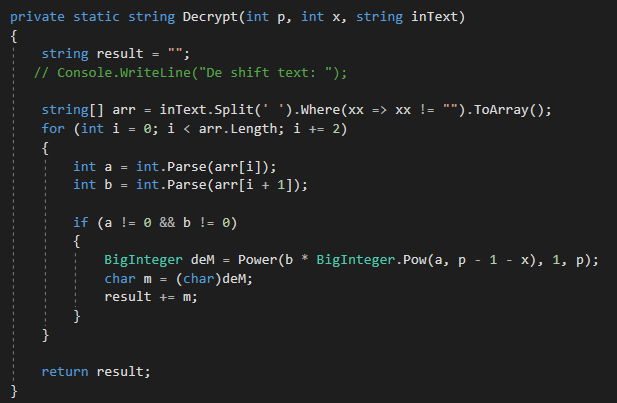


Рисунок 1.6 – Функция Decrypt()

Итог выполнения программы представлен на рисунке 1.7

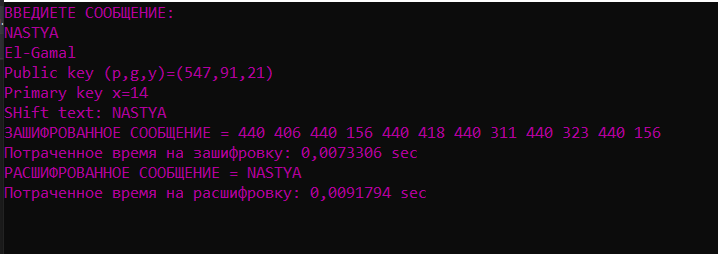


Рисунок 1.7 – Итог выполнения алгоритма Эль-Гамаля