Министерство образования республики Молдова

Технический Университет Молдовы

Департамент Программной Инженерии и Автоматики

Отчёт

Лабораторная работа №5

По предмету: Metode criptografice de protective a informatiei  
Тема: Criptografia cu chei publice

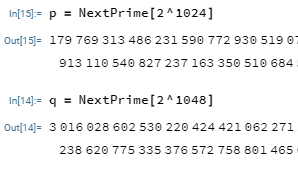
Выполнил студ. гр. SI-202 Абабий Эдуард

Проверил Aureliu Zgureanu

Кишинёв – 2023

**Задание 2.1.**

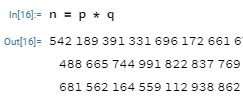
Используя платформу wolframalpha.com или приложение Wolfram Mathematica, сгенерируйте ключи, зашифруйте и расшифруйте сообщение m = Имя Фамилия, применяя алгоритм RSA. Значение n должно быть не менее 2048 бит



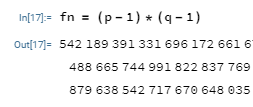
Сгенерировали 2 простых числа p и q

Количество бит числа p – 1025

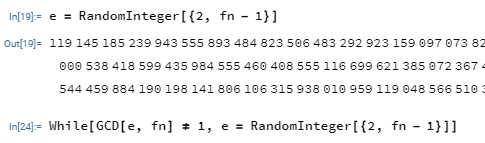
Количество бит числа q – 1049



Вычислили n, количество бит которого 2073



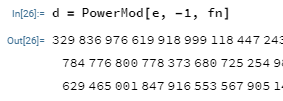
Затем вычислили функцию Эйлера, то есть количество простых чисел от 1 до n-1 , которые взаимно просты с n



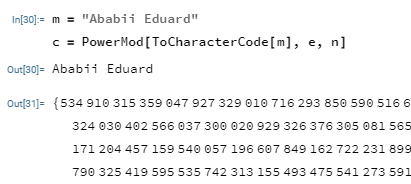
Необходимо выбрать открытую экспоненту e, которая является целым числом, взаимно простым с функцией Эйлера fn(n). Для этого выбирается случайное число e в диапазоне от 2 до fn(n) - 1. Затем происходит проверка на взаимную простоту чисел e и fn(n), используя функцию GCD, которая вычисляет наибольший общий делитель двух чисел.

Если GCD(e, phi(n)) не равно 1, то это означает, что e не является взаимно простым с fn(n), и выбирается новое случайное значение e и происходит повторная проверка.

Таким образом, цикл While генерирует случайное число e до тех пор, пока оно не будет взаимно простым с fn(n).



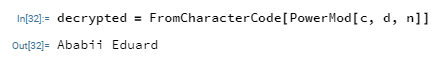
Секретная экспонента d вычисляется с использованием открытой экспоненты e и функции Эйлера fn(n). Она используется для расшифровки зашифрованного сообщения



В данном случае используется функция ToCharacterCode, которая преобразует строку m в список числовых значений, соответствующих кодам символов в ASCII.

Затем каждое числовое значение символа в сообщении m возводится в степень e по модулю n с помощью функции PowerMod. Результатом является зашифрованный список числовых значений c.

После этого зашифрованный список числовых значений может быть отправлен получателю, который сможет расшифровать его, используя свою секретную экспоненту d и свой секретный ключ (n, d).



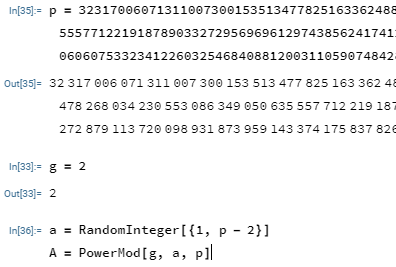
Для расшифровки каждого элемента списка c, получатель возводит его в степень d по модулю n с помощью функции PowerMod. Результатом будет список числовых значений decrypted, которые соответствуют исходной строке m.

Чтобы получить исходную строку из списка числовых значений, используется функция FromCharacterCode, которая преобразует список числовых значений в соответствующую строку символов с помощью заданной кодировки (например, ASCII или Unicode). Таким образом, переменная decrypted будет содержать расшифрованное сообщение m.

**Задание 2.2.**

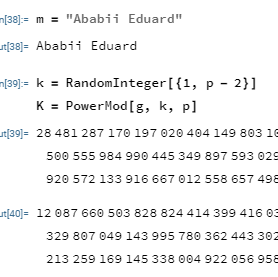
Используя платформу wolframalpha.com или приложение Wolfram Mathematica, сгенерируйте ключи, зашифруйте и расшифруйте сообщение m = Имя Фамилия, применяя алгоритм Эль-Гамаля. Значения p и генератора заданы ниже.

p=3231700607131100730015351347782516336248805713348907517458843413926 980683413621000279205636264016468545855635793533081692882902308057347 262527355474246124574102620252791657297286270630032526342821314576693 141422365422094111134862999165747826803423055308634905063555771221918 789033272956969612974385624174123623722519734640269185579776797682301 462539793305801522685873076119753243646747585546071504389684494036613 049769781285429595865959756705128385213278446852292550456827287911372 009893187395914337417583782600027803497319855206060753323412260325468 4088120031105907484281003994966956119696956248629032338072839127039, care are 2048 biți și generatorul g=2.

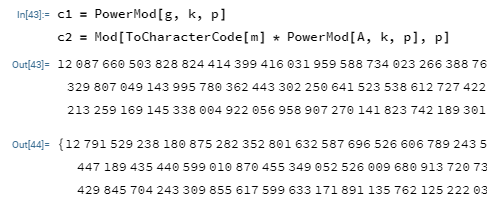


Генерация закрытого ключа a и соответствующего ему открытого ключа A. Закрытый ключ обычно представляет собой случайно выбираемое число в определенном диапазоне (в данном случае в диапазоне от 1 до p-2), который затем используется для вычисления открытого ключа.

В формуле A = PowerMod[g, a, p] мы берем число g (в данном случае 2) и возводим его в степень a по модулю p, чтобы получить значение открытого ключа A. Затем A будет использован в дальнейшем для зашифрования сообщения.



Выбор случайного числа k из диапазона [1, p-2], которое используется для вычисления открытого ключа K. Открытый ключ K является результатом возведения числа g (генератора) в степень k по модулю p.

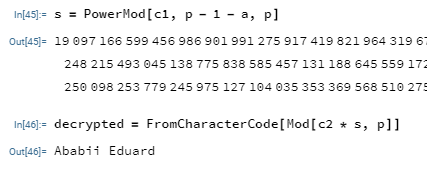


Здесь мы шифруем сообщение m, используя публичный ключ A и сессионный ключ k.

Переменная c1 вычисляется как g в степени k по модулю p, т.е. это первая часть зашифрованного сообщения, которую отправитель передает получателю.

Переменная c2 вычисляется как массив кодов символов сообщения m, умноженный на A в степени k по модулю p. Далее происходит взятие остатка от деления этого произведения на p. Таким образом, c2 представляет собой вторую часть зашифрованного сообщения.

В итоге, пара (c1, c2) будет отправлена получателю в качестве зашифрованного сообщения.



Сначала мы вычисляем секретный ключ с помощью операции возведения в степень по модулю p:

s = PowerMod[c1, p - 1 - a, p]

Здесь мы используем значение c1, которое было вычислено при зашифровывании сообщения и передано получателю. Мы также используем значение a, которое было вычислено отправителем и является частью его секретного ключа.

Затем мы используем полученный секретный ключ s, чтобы расшифровать значение c2, которое также было передано получателю:

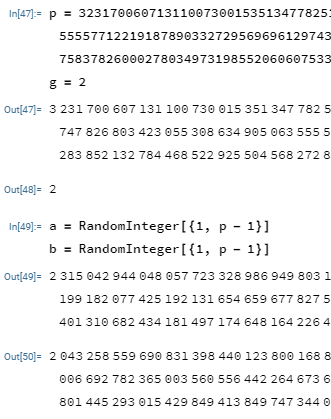
decrypted = FromCharacterCode[Mod[c2 \* s, p]]

Значение c2 было вычислено отправителем, а затем зашифровано с помощью открытого ключа получателя. Здесь мы используем операцию умножения и взятия остатка от деления на модуль p, чтобы дешифровать сообщение. Затем мы используем функцию FromCharacterCode, чтобы получить исходную строку из расшифрованных значений.

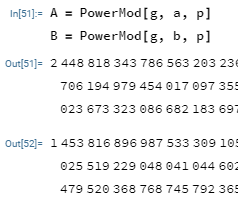
**Задание 3.**

Используя платформу wolframalpha.com или приложение Wolfram Mathematica, выполнить обмен ключами Диффи-Хелмана между Алисой и Бобом, которые используют алгоритм AES с 256-битным ключом. Секретные числа a и b должны быть выбраны случайным образом в соответствии с требованиями алгоритма. Значения p и генератора заданы ниже

p=3231700607131100730015351347782516336248805713348907517458843413926 980683413621000279205636264016468545855635793533081692882902308057347 262527355474246124574102620252791657297286270630032526342821314576693 141422365422094111134862999165747826803423055308634905063555771221918 789033272956969612974385624174123623722519734640269185579776797682301 462539793305801522685873076119753243646747585546071504389684494036613 049769781285429595865959756705128385213278446852292550456827287911372 009893187395914337417583782600027803497319855206060753323412260325468 4088120031105907484281003994966956119696956248629032338072839127039, care are 2048 biți și generatorul g=2.



Переменные a и b - секретные случайные числа, которые выбирают Алиса и Боб соответственно. Они являются приватными ключами, которые будут использоваться для вычисления общего ключа, который будут использовать для шифрования сообщений с помощью алгоритма AES с 256-битным ключом.

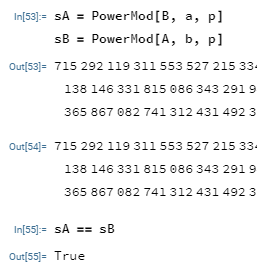


Далее, после того, как Алиса и Боб выбрали свои секретные числа, они вычисляют соответствующие открытые числа.

Алиса вычисляет своё открытое число, используя формулу A = g^a mod p, где g - это генератор, a - это её секретное число, а p - это большое простое число, известное обоим участникам.

Боб вычисляет своё открытое число, используя аналогичную формулу B = g^b mod p, где b - это его секретное число.

Таким образом, Алиса и Боб обменялись открытыми числами A и B, не раскрывая при этом свои секретные числа.



После того, как Алиса и Боб выбрали свои секретные числа, они вычисляют открытые числа, используя параметры g и p и свои секретные числа a и b, соответственно.

Далее, они передают эти открытые числа друг другу. Затем каждый из них вычисляет общий секретный ключ, используя открытое число, полученное от другого и своё секретное число.

После этого они должны получить одинаковый общий секретный ключ sA, который означает, что процедура Диффи-Хеллмана прошла успешно и они могут использовать sA для защищенного обмена сообщениями. Сравнение sA == sB проверяет, что общий секретный ключ sA у Алисы и Боба совпадает.

**Вывод:**В данной лабораторной работе я научился зашифровывать сообщения с помощью алгоритма RSA, Эль-Гамаля и создавать пару открытых и закрытых ключей для обмена.