Криптографические средства

Лабораторная работа №1

Вариант 1.7

Выполнили студенты группы 22307

Никаноров Роман и Харичев Евгений

ПетрГУ 2022

Формулировка задания

Напишите программу формирования цифровой подписи алгоритмом RSA. Рекомендуется использовать библиотеку для работы с длинными числами. В случае применения этой библиотеки разрешается использовать функции сложения, вычитания, умножения, целочисленного деления, вычисления остатка от деления. Функции возведения числа в степень, нахождения наибольшего общего делителя, обратного элемента в мультипликативной группе вычетов, генерации простого числа реализовать самостоятельно. Для ускорения вычислений использовать китайскую теорему об остатках. Выполняемые функции программы:

1. генерация пары открытый/закрытый ключ, число e должно генерироваться после вычисления p и q;
2. получения цифровой подписи для сообщения (целого числа);
3. проверки цифровой подписи для данного сообщения (целого числа).

Описание метода решения

1. Генерируем случайным образом большое простое число :
   1. Получаем случайное целое число из отрезка .
   2. Проверяем его на простоту использую простые числа, меньшие 1000.
   3. Проверяем его на простоту тестом Рабина Миллера.
   4. Повторяем 1.1-1.3, пока не получим простое число.
2. Генерируем тем же способом второе большое простое число , не равное .
3. Определяем криптомодуль .
4. Подсчитываем значение функции Эйлера .
5. Подбираем небольшое нечетное число , взаимно простое с :
   1. Начинаем перебирать все нечетные числа, начиная с 3.
   2. Вычисляем НОД() по алгоритму Евклида.
   3. Проверяем равенство НОД() = 1.
   4. Берем следующее нечетное число и повторяем 5.2 ­­­­­ 5.3, пока не выполнится равенство.
6. Вычисляем , где вычисляется методом расширенного алгоритма Евклида из равенства .
7. Составляем открытый ключ – пару .
8. Составляем закрытый ключ – пару .
9. Получаем цифровую подпись для сообщения (целого числа):

.

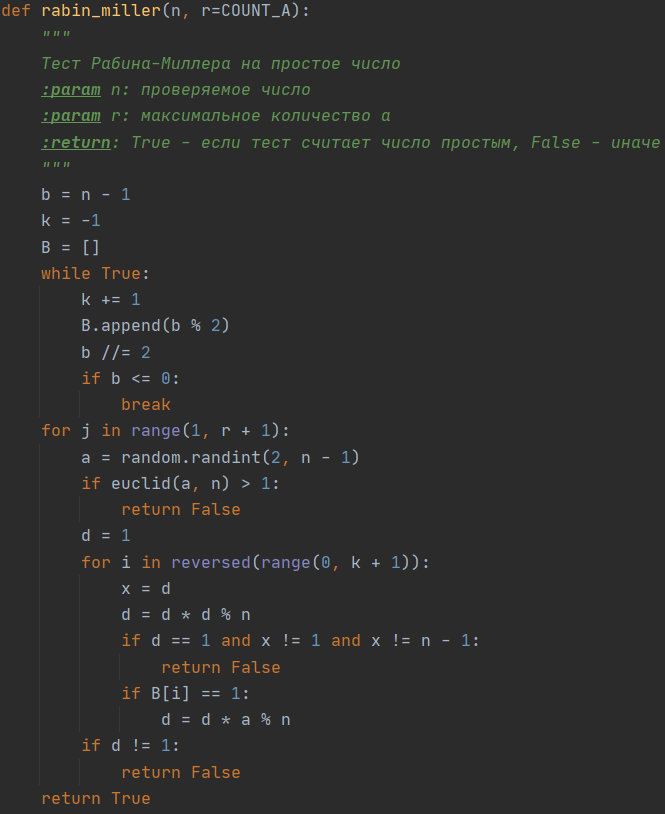
* 1. Возведение в степень и вычисление остатка выполняем методом повторного возведения в квадрат.

1. Проверяем цифровую подпись для данного сообщения по равенству , где:

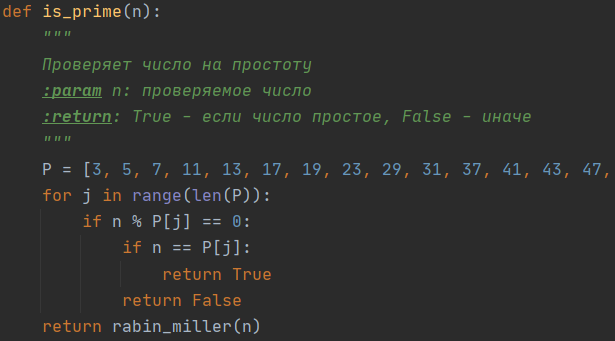
*.*

Примеры кода программ

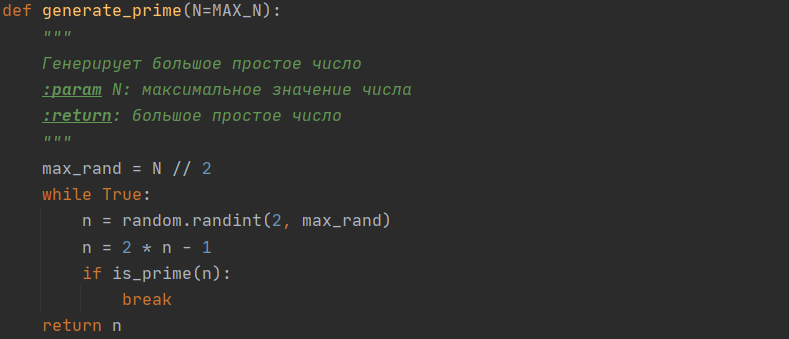
Тест Рабина-Миллера на простое число:



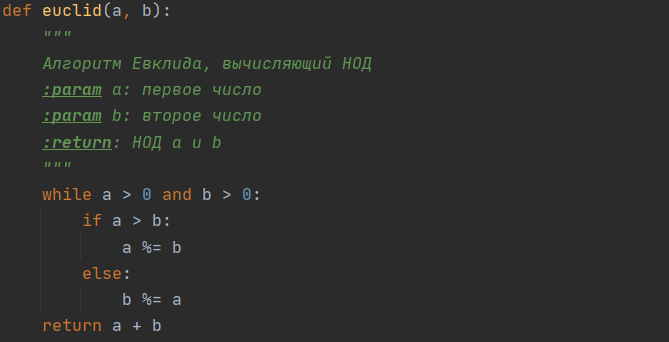
Проверка на простое число:



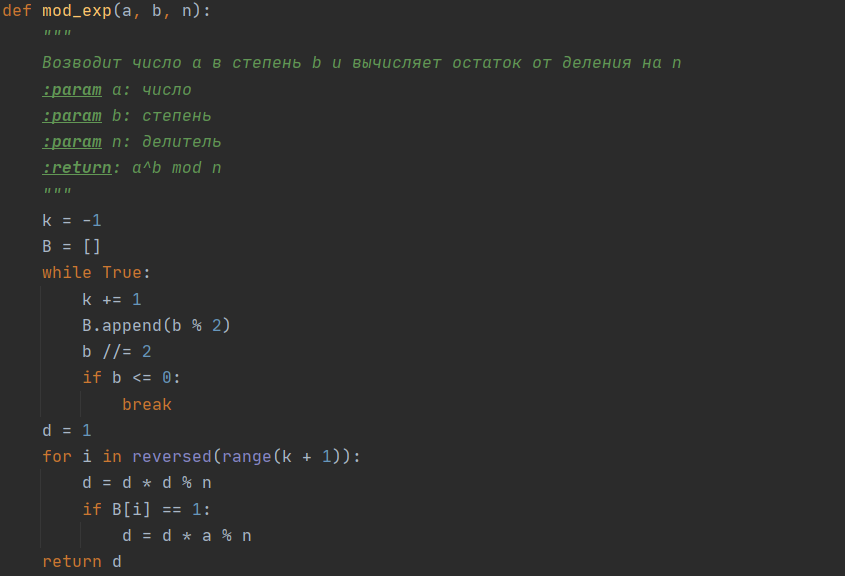
Генерация большого простого числа:



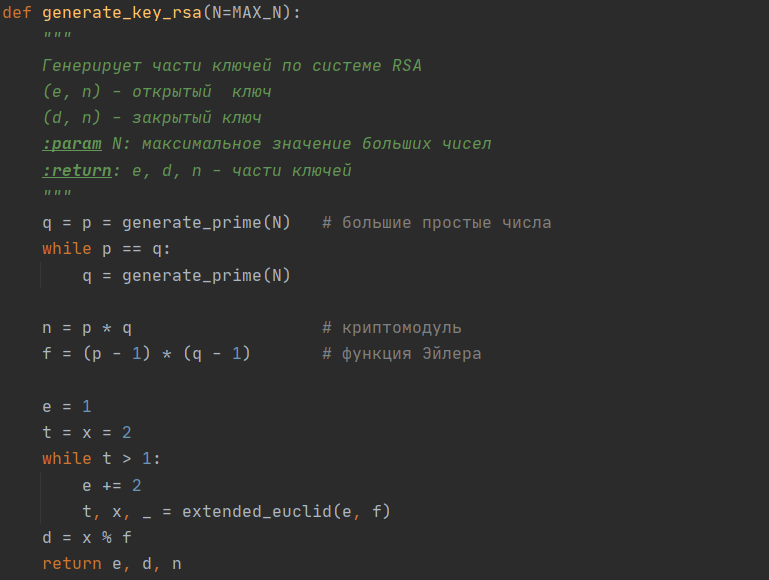
Алгоритм Евклида, вычисляющий НОД:



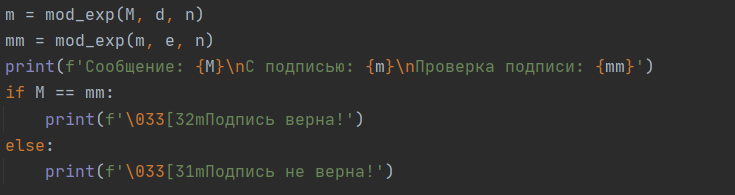
Метод повторного возведения в квадрат:



Метод для генерации частей ключей по системе RSA:



Получение цифровой подписи и её проверка:



Тестовые данные

Примеры работы программы для сообщения M = 123456789, максимальной длине чисел в 25 символов и количестве проверок в тесте Рабина-Миллера равном 100:

