Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОССУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОННИКИ

КАФЕДРА ИНФОРМАТИКИ

Отчёт по лабораторной работе №4

По дисциплине «Методы защиты информации»

По теме «Ассиметричная криптография. Алгоритм Эль-Гамаля»

Выполнил:

студент гр. 653502

Максименко П.В.

Проверил:

Артемьев В. С.

Минск 2019

1. Постановка зада4и

Написать программу, реализующую алгоритм Эль-Гамаля шифрования данных.

2. Теорети4еская 4асть

Схема Эль-Гамаля (Elgamal) — [криптосистема](https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) с открытым ключом, основанная на трудности вычисления [дискретных логарифмов](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D0%BA%D1%80%D0%B5%D1%82%D0%BD%D1%8B%D0%B9_%D0%BB%D0%BE%D0%B3%D0%B0%D1%80%D0%B8%D1%84%D0%BC) в [конечном поле](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D0%B5%D1%87%D0%BD%D0%BE%D0%B5_%D0%BF%D0%BE%D0%BB%D0%B5). Криптосистема включает в себя алгоритм шифрования и алгоритм цифровой подписи. Схема Эль-Гамаля лежит в основе бывших стандартов [электронной цифровой подписи](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%86%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D0%B0%D1%8F_%D0%BF%D0%BE%D0%B4%D0%BF%D0%B8%D1%81%D1%8C) в [США](https://ru.wikipedia.org/wiki/%D0%A1%D0%A8%D0%90) ([DSA](https://ru.wikipedia.org/wiki/DSA)) и [России](https://ru.wikipedia.org/wiki/%D0%A0%D0%BE%D1%81%D1%81%D0%B8%D1%8F).

3. Блок-схема

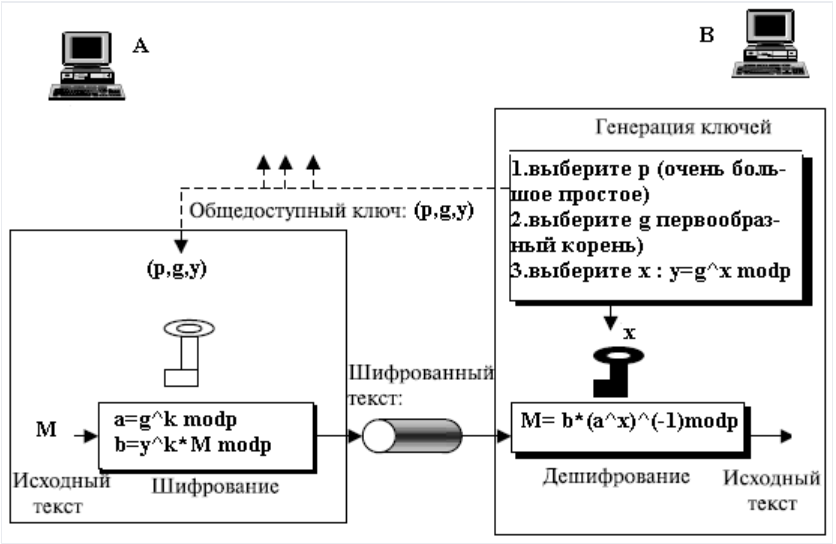


Рис 3.1. Схема работы алгоритма.

4. Ход программы

, где data\_start – 4исло, которое необходимо зашифровать, а p необходимо для генерации клю4ей

Рис 4.1. Данные, поступающие на вход



Рис 4.2. Нахождение первообразного корня



Рис 4.3. Создание клю4ей для шифрации



Рис 4.4. Зашифрованные данные

5. Код программы

int ar(int p)

{

int help[10] = {0,0,0,0,0,0,0,0,0,0};

int count = 0;

for (int i = 2; i < p; i++) {

if ((p - 1) % i == 0)

{

help[count] = i;

count++;

}

}

int ret = 0;

for (int i = 2; i < p; i++) {

int flag = 0;

for (int j = 0; j < 100; j++) {

if ((help[j] != 0) && (int(pow(i, help[j]) - 1) % p == 0))

{

flag == 1;

}

}

if (flag == 0)

{

ret = i;

break;

}

}

return ret;

}

float\* ElGamal()

{

int data = 10; // data < p

std::cout << "data\_start = " << data << std::endl;

// keygen

int p = 11;

int g = ar(p); // antiderivative root p

int x = 8; // 1 < x < p - 1

std::cout << "p = " << p << std::endl;

std::cout << "antiderivative root p = " << g << std::endl;

float y = pow(g, x);

y = int(y) % p;

float public\_key = y;

int private\_key = x;

std::cout << "public\_key = " << y << std::endl;

std::cout << "private\_key = " << x << std::endl;

int k = 4; // 1 < k < p - 1

float a[2];

a[0] = pow(g, k);

a[0] = int(a[0]) % p;

a[1] = pow(y, k);

a[1] = int(a[1] \* data) % p;

std::cout << "ciphertext1 = " << a[0] << std::endl;

std::cout << "ciphertext2 = " << a[1] << std::endl;

return a;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

// lab4

// El Gamal

float\* numbs = ElGamal();

return 0;

}

6. Вывод

В ходе выполнения лабораторной работы был успешно реализован алгоритм Эль-Гамаля. Были выявлены его сильные и слабые стороны, а также возможности улу4шения. Криптостойкость алгоритма:

Y = g^x mod p.

Также существует возможность работы алгоритма в режиме подписи.