МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ государственное БЮДЖЕТНОЕ

образовательное учреждение

высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра защиты информации



**ОТЧЁТ**

**по лабораторной работе № 3**

**«***Программирование циклического вычислительного процесса***»**

**по дисциплине: «***Программирование***»**

Выполнил:Проверил:

Студент гр. «АБ-121», «АВТФ» *доцент кафедры ЗИ*

*Втюрин Александр Романович Архипова А. Б.*

«4» июня 2022г«\_\_\_» \_\_\_\_\_\_ 2022 г.

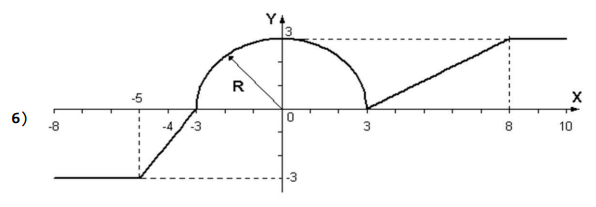
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись) (подпись)

Новосибирск 2022

***Цели и задачи работы:*** изучение циклических алгоритмов, операторов цикла, программирование циклического вычислительного процесса.

**Задание к работе:** реализовать циклический вычислительный процесс. Самостоятельно решить задачи в соответствии с индивидуальным вариантов.

**Задание 1.** Вычислить и вывести на экран или в файл в виде таблицы значения функции, заданной графически, на интервале от Xнач до Xкон с шагом dx. Интервал и шаг задать таким образом, чтобы проверить все ветви программы. Таблица должна иметь заголовком и шапку.

**Задание 2.** Реализовать ax modp Сравнения по модулю простого числа через теорему Ферма и свойства сравнений.

**Задание 3.** Реализовать обобщенный алгоритм Евклида для вычисления с\*d mod m=1.

**Задание 4.** Реализовать расширенный алгоритм Евклида для вычисления взаимообратного числа с-1 mod m=d.

**Задание 5.** Написать программу, использующую алгоритм шифрования данных для преобразования исходного текста.

Написать программу, использующую крипто протоколы вида.

Вариант 1 - Диффи-Хеллмана

**Задание 6\*.** Тесты на простоту.

1) Построить таблицу простых чисел, меньших 500, с помощью решета Эратосфена. С использованием этой таблицы:

а) реализовать процедуру получения простых чисел заданной длины на основе теста Миллера;

б) реализовать процедуру получения простых чисел заданной длины на основе теста Поклингтона;

в) реализовать процедуру генерации простых чисел заданной длины ГОСТ Р 34.10-94.

**Задание 7.** Найти последнюю цифру «трехэтажного числа». Например, 37^8

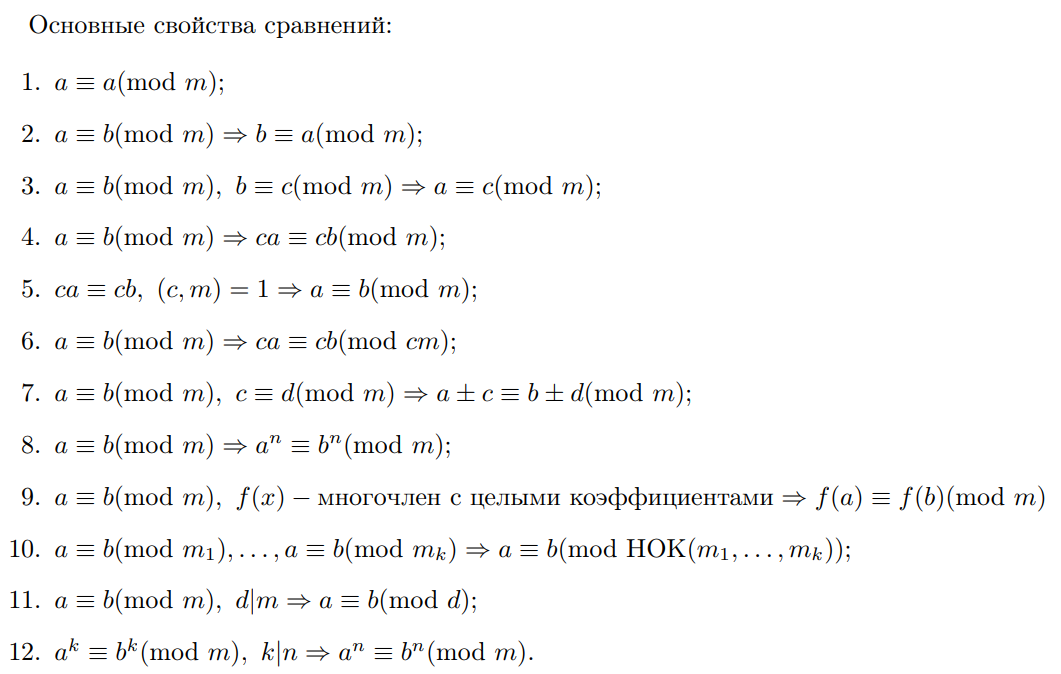
**Теоретическая часть:**

**Задание 2.** Реализовать ax modp Сравнения по модулю простого числа через теорему Ферма и свойства сравнений.

(1) Пусть a, b ∈ Z, m ∈ N. Говорят, что a сравнимо с b по модулю m, если (a − b) делится на m. В этом случае будем писать a ≡ b(mod m).

(2) Если р - простое число, и а - целое число, взаимно простое с р, то

а(p-1) mod p ≡ l.



**Задание 3.** Реализовать обобщенный алгоритм Евклида для вычисления с\*d mod m=1.

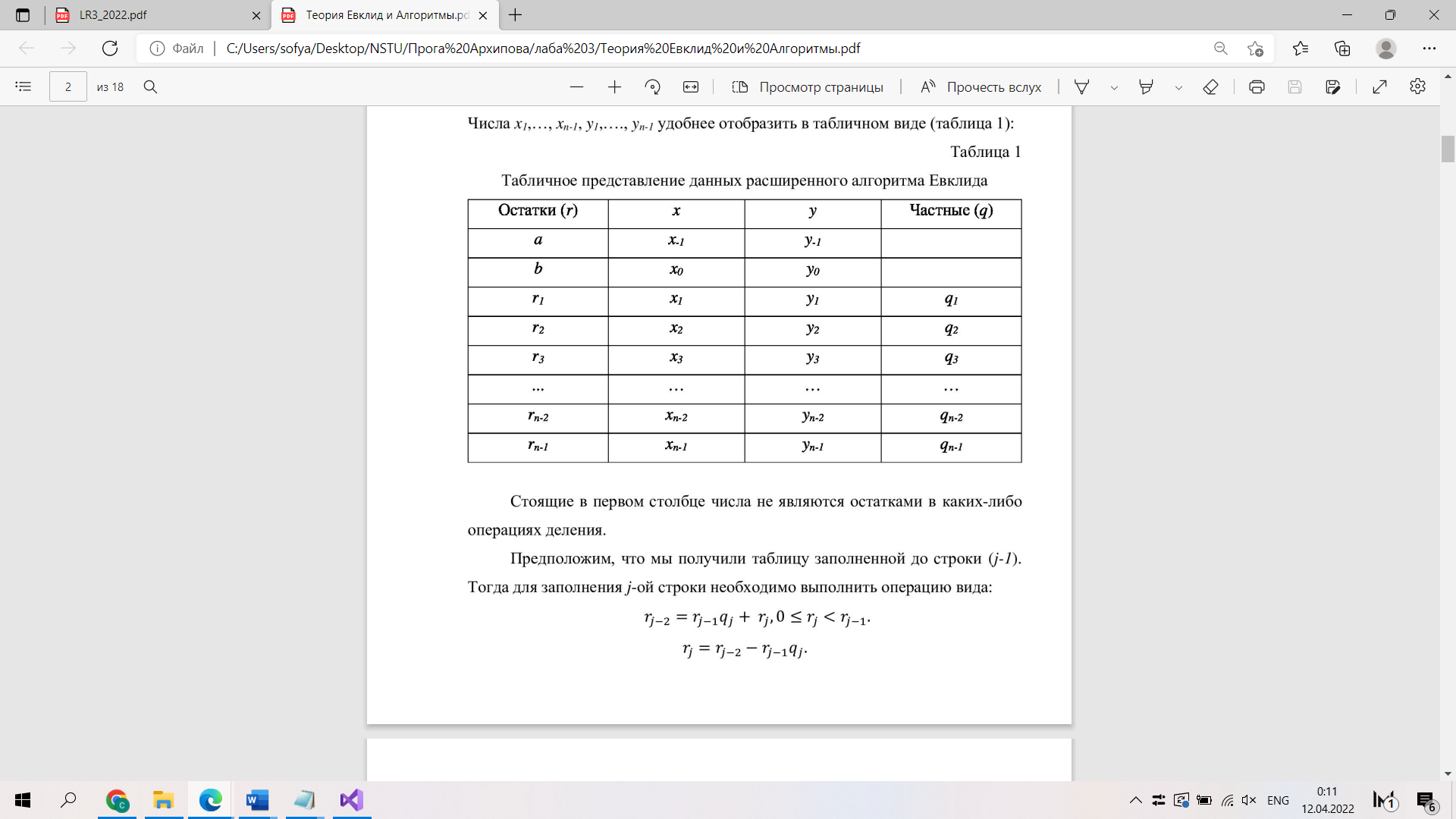
**Теорема**. Пусть а и b — два целых положительных числа. Тогда существуют целые (не обязательно положительные) числа х и у, такие, что

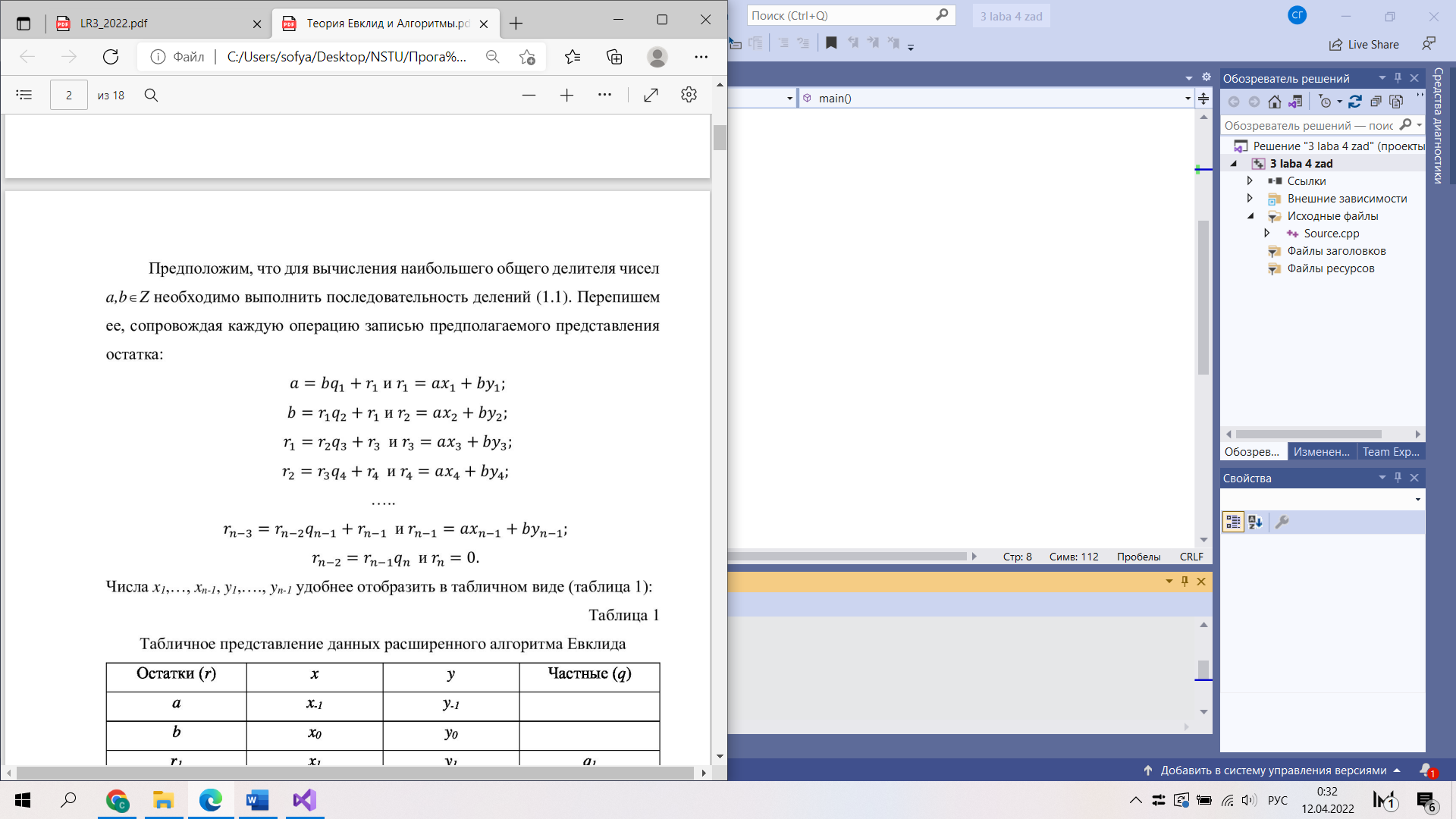
ах + by = НОД(a,b).

Обобщенный *алгоритм* Евклида служит для отыскания НОД(a, b) и х, у, удовлетворяющих записанному выше уравнению.

**Задание 4.** Реализовать расширенный алгоритм Евклида для вычисления взаимообратного числа с-1 mod m=d.

Расширенный алгоритм Евклида вычисляет кроме НОД целых чисел, еще и коэффициенты соотношения Безу (линейная комбинация с целыми коэффициентами). Широко используется в криптографии.





**Задание 5.** Написать программу, использующую алгоритм шифрования данных для преобразования исходного текста.

Написать программу, использующую крипто протоколы вида.

Вариант 1 - Диффи-Хеллмана

**Протокол Диффи — Хеллмана** — криптографический протокол, позволяющий двум и более сторонам получить общий секретный ключ, используя незащищенный от прослушивания канал связи. Полученный ключ используется для шифрования дальнейшего обмена с помощью алгоритмов симметричного шифрования.

**Практическая часть:**

**С++**

#include <iostream>

#include <math.h>

#include <stdio.h>

#include <locale>

#include <cstdint>

#include <vector>

#include <string>

using namespace std;

void Task1 ()

{

cout << " " << "X" << " " << "Y" << endl;

for (int x = -8, y = -3; x <= -5; x++)

{

cout << x << " " << y << endl;

}

for (double x = -5, y = -3.0; x <= -3; x++)

{

y = 1.5 \* x + 4.5;

cout << x << " " << y << endl;

}

for (double x = -3, y = 0; x <= 3 ; x++)

{

y += x;

cout << x << " " << y << endl;

}

for (double x = 3, y = 0; x <= 8; x++)

{

y = 0.27 \* x + 0.82;

cout << x << " " << y << endl;

}

for (int x = 8, y = 3; x <= 10; x++)

{

cout << x << " " << y << endl;

}

}

int Task2 (const int base, const int exponent, const int mod)

{

int pow = 1;

for (int i = 0; i < exponent; i++)

{

pow \*= base;

pow %= mod;

}

return pow;

}

void Task3 ()

{

int a, b, i = 1, r = 1, q;

cout << "Введите большее число: " << endl;

cin >> a;

cout << "Введите меньшее число: " << endl;

cin >> b;

while (i == 1)

{

if (r != 0)

{

r = a % b;

q = a / b;

a = b;

b = r;

}

else

{

cout << "НОД = " << a << endl;

i = 0;

}

}

}

void Task4 ()

{

int a, b, i = 1, r = 1, q, x1 = 1, y1 = 0, x2 = 0, y2 = 1, x3, y3, a1; //зададим x1 и y1 начальные значени

cout << "введите большое число: " << endl;

cin >> a;

cout << "Введите меньшее число: " << endl;

cin >> b;

cout << "r\t\tx\t\ty\t\tq" << endl;

cout << a << "\t\t" << x1 << "\t\t" << y1 << "\t\t" << endl;

cout << b << "\t\t" << x2 << "\t\t" << y2 << endl;

while (i == 1)

{

if (r != 0)

{

r = a % b;

q = a / b;

x3 = x1 - q \* x2;

y3 = y1 - q \* y2;

x1 = x2;

x2 = x3;

y1 = y2;

y2 = y3;

if (r != 0) cout << r << "\t\t" << x3 << "\t\t" << y3 << "\t\t" << q << endl;

else cout << r << "\t\t" << "---" << "\t\t" << "---" << "\t\t" << q << endl;

a = b;

b = r;

}

else

{

cout << "НОД = " << a << endl;

i = 0;

}

}

}

void Task5 ()

{

int g, p, a, b, Xa = 1, Xb = 1, Ya = 1, Yb = 1;

cout << "Придумайте 2 случайных числа: ";

cin >> g;

cin >> p;

cout << "Придумайте секретное число для Алисы: ";

cin >> a;

cout << "Придумайте секретное число для Боба: ";

cin >> b;

for (int i = 0; i < a; i++)

{

Ya \*= g;

Ya %= p;

}

cout << "Публичный ключ Алисы: " << Ya << endl;

for (int i = 0; i < b; i++)

{

Yb \*= g;

Yb %= p;

}

cout << "Публичный ключ Боба: " << Yb << endl;

for (int i = 0; i < a; i++)

{

Xa \*= Yb;

Xa %= p;

}

cout << "Приватный ключ Алисы: " << Xa << endl;

for (int i = 0; i < b; i++)

{

Xb \*= Ya;

Xb %= p;

}

cout << "Приватный ключ Боба: " << Xb << endl;

}

void Task7 ()

{

int x = 3, r = 7, r1 = 8, ans = 0, itog\_st = 0;

int t = x % 10;

int t1 = r % 10;

int t2 = r1 % 10;

//Числа с очевидными последними цифрами

if (t == 0) cout << "Последняя цифра числа: 0";

else if (t == 1) cout << "Последняя цифра числа: 1";

else if (t == 5) cout << "Последняя цифра числа: 5";

else if (t == 6) cout << "Последняя цифра числа: 6";

else {

for (int i = 1; i <= r1; i++)

{ //находим последнюю цифру итоговой степени

itog\_st = r % 10;

r = t1 \* itog\_st;

}

if (itog\_st == 0) cout << "Последняя цифра числа: 1";

else

{

for (int j = 1; j <= itog\_st; j++)

{

ans = x % 10;

x = t \* ans;

}

cout << "Последняя цифра числа: " << ans;

}

}

}

int main ()

{

setlocale(LC\_ALL, "Russian");

cout << "Task 1:" << endl;

Task1 ();

cout << "Task 2:" << endl;

int base, exponent, mod, pow;

cout << "Enter a: ";

cin >> base;

cout << "Enter x: ";

cin >> exponent;

cout << "Enter p:";

cin >> mod;

cout << Task2 (base, exponent, mod) << endl;

cout << "Task 3:" << endl;

Task3 ();

cout << "Task 4:" << endl;

Task4();

cout << "Task 5:" << endl;

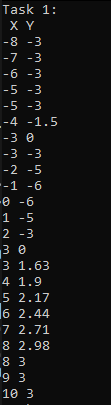
Task5();

cout << "Task 7:" << endl;

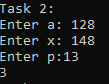
Task7();

}

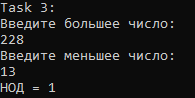
Task1



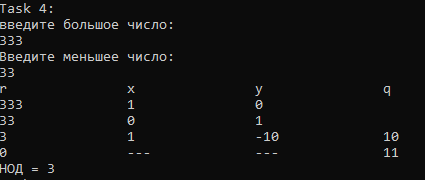
Task2



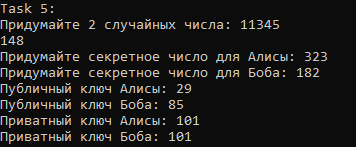
Task3



Task4



Task5



Task7



**С#**

using System;

public static class Lab3

{

public static void Task1()

{

Console.Write(" ");

Console.Write("X");

Console.Write(" ");

Console.Write("Y");

Console.Write("\n");

for (int x = -8, y = -3; x <= -5; x++)

{

Console.Write(x);

Console.Write(" ");

Console.Write(y);

Console.Write("\n");

}

for (double x = -5, y = -3.0; x <= -3; x++)

{

y = 1.5 \* x + 4.5;

Console.Write(x);

Console.Write(" ");

Console.Write(y);

Console.Write("\n");

}

for (double x = -3, y = 0; x <= 3 ; x++)

{

y += x;

Console.Write(x);

Console.Write(" ");

Console.Write(y);

Console.Write("\n");

}

for (double x = 3, y = 0; x <= 8; x++)

{

y = 0.27 \* x + 0.82;

Console.Write(x);

Console.Write(" ");

Console.Write(y);

Console.Write("\n");

}

for (int x = 8, y = 3; x <= 10; x++)

{

Console.Write(x);

Console.Write(" ");

Console.Write(y);

Console.Write("\n");

}

}

public static int Task2(int @base, int exponent, int mod)

{

int pow = 1;

for (int i = 0; i < exponent; i++)

{

pow \*= @base;

pow %= mod;

}

return pow;

}

public static void Task3()

{

int a;

int b;

int i = 1;

int r = 1;

int q;

Console.Write("Enter a larger number: ");

Console.Write("\n");

a = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

Console.Write("Enter a smaller number: ");

Console.Write("\n");

b = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

while (i == 1)

{

if (r != 0)

{ //a=bq+r

r = a % b;

q = a / b;

a = b;

b = r;

}

else

{

Console.Write("NOD = ");

Console.Write(a);

Console.Write("\n");

i = 0;

}

}

}

public static void Task4()

{

int a; //зададим x1 и y1 начальные значени

int b;

int i = 1;

int r = 1;

int q;

int x1 = 1;

int y1 = 0;

int x2 = 0;

int y2 = 1;

int x3;

int y3;

Console.Write("Enter a larger number: ");

Console.Write("\n");

a = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

Console.Write("Enter a smaller number: ");

Console.Write("\n");

b = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

Console.Write("r\t\tx\t\ty\t\tq");

Console.Write("\n");

Console.Write(a);

Console.Write("\t\t");

Console.Write(x1);

Console.Write("\t\t");

Console.Write(y1);

Console.Write("\t\t");

Console.Write("\n");

Console.Write(b);

Console.Write("\t\t");

Console.Write(x2);

Console.Write("\t\t");

Console.Write(y2);

Console.Write("\n");

while (i == 1)

{

if (r != 0)

{

r = a % b;

q = a / b;

x3 = x1 - q \* x2;

y3 = y1 - q \* y2;

x1 = x2;

x2 = x3;

y1 = y2;

y2 = y3;

if (r != 0)

{

Console.Write(r);

Console.Write("\t\t");

Console.Write(x3);

Console.Write("\t\t");

Console.Write(y3);

Console.Write("\t\t");

Console.Write(q);

Console.Write("\n");

}

else

{

Console.Write(r);

Console.Write("\t\t");

Console.Write("---");

Console.Write("\t\t");

Console.Write("---");

Console.Write("\t\t");

Console.Write(q);

Console.Write("\n");

}

a = b;

b = r;

}

else

{

Console.Write("NOD = ");

Console.Write(a);

Console.Write("\n");

i = 0;

}

}

}

public static void Task5()

{

int g;

int p;

int a;

int b;

int Xa = 1;

int Xb = 1;

int Ya = 1;

int Yb = 1;

Console.Write("Come up with 2 random numbers: ");

g = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

p = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

Console.Write("Come up with a secret number for Alice: ");

a = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

Console.Write("Come up with a secret number for Bob: ");

b = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

for (int i = 0; i < a; i++)

{

Ya \*= g;

Ya %= p;

}

Console.Write("Alice's Public key: ");

Console.Write(Ya);

Console.Write("\n");

for (int i = 0; i < b; i++)

{

Yb \*= g;

Yb %= p;

}

Console.Write("Bob's Public key: ");

Console.Write(Yb);

Console.Write("\n");

for (int i = 0; i < a; i++)

{

Xa \*= Yb;

Xa %= p;

}

Console.Write("Alice's Private key: ");

Console.Write(Xa);

Console.Write("\n");

for (int i = 0; i < b; i++)

{

Xb \*= Ya;

Xb %= p;

}

Console.Write("Приватный ключ Боба: ");

Console.Write(Xb);

Console.Write("\n");

}

public static void Task7()

{

int x = 3;

int r = 7;

int r1 = 8;

int ans = 0;

int itog\_st = 0;

int t = x % 10;

int t1 = r % 10;

int t2 = r1 % 10;

//Числа с очевидными последними цифрами

if (t == 0)

{

Console.Write("Последняя цифра числа: 0");

}

else if (t == 1)

{

Console.Write("Последняя цифра числа: 1");

}

else if (t == 5)

{

Console.Write("Последняя цифра числа: 5");

}

else if (t == 6)

{

Console.Write("Последняя цифра числа: 6");

}

else

{

for (int i = 1; i <= r1; i++)

{ //находим последнюю цифру итоговой степени

itog\_st = r % 10;

r = t1 \* itog\_st;

}

if (itog\_st == 0)

{

Console.Write("The last digit of the number: 1");

}

else

{

for (int j = 1; j <= itog\_st; j++)

{

ans = x % 10;

x = t \* ans;

}

Console.Write("The last digit of the number: ");

Console.Write(ans);

}

}

}

static int Main()

{

Console.Write("Task 1:");

Console.Write("\n");

Task1();

Console.Write("Task 2:");

Console.Write("\n");

int @base;

int exponent;

int mod;

Console.Write("Enter a: ");

@base = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

Console.Write("Enter x: ");

exponent = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

Console.Write("Enter p:");

mod = int.Parse(ConsoleInput.ReadToWhiteSpace(true));

Console.Write(Task2(@base, exponent, mod));

Console.Write("\n");

Console.Write("Task 3:");

Console.Write("\n");

Task3();

Console.Write("Task 4:");

Console.Write("\n");

Task4();

Console.Write("Task 5:");

Console.Write("\n");

Task5();

Console.Write("Task 7:");

Console.Write("\n");

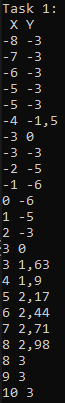
Task7();

return 0;

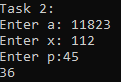
}

}

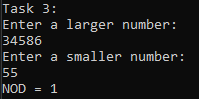
Task1



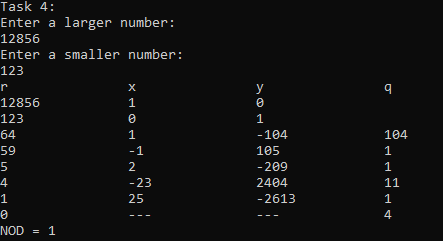
Task2



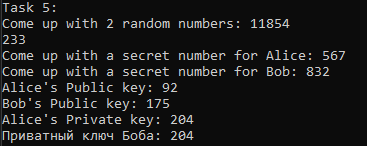
Task3



Task4



Task5



Task7



Вывод:

В ходе проделанной работы изучены циклические алгоритмы, операторы циклов, программирование циклического вычислительного процесса. Реализовала циклический вычислительный процесс. Задачи решены в соответствии с индивидуальным вариантом. Поставленные задачи реализованы на двух языках программирования: C++ и C#.