

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**

**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**(ДГТУ)**

Факультет «Информатика и вычислительная техника»

Кафедра «Кибербезопасность информационных систем»

Лабораторная работа №5

по дисциплине: «Теоретико-числовые методы в криптографии»

на тему: «Символы Лежандра и Якоби.»

Выполнил обучающийся гр. ВКБ42

Михайлов Александр Сергеевич

          Проверила:

Ст. пр. Артамонова Е.А.

Ростов-на-Дону

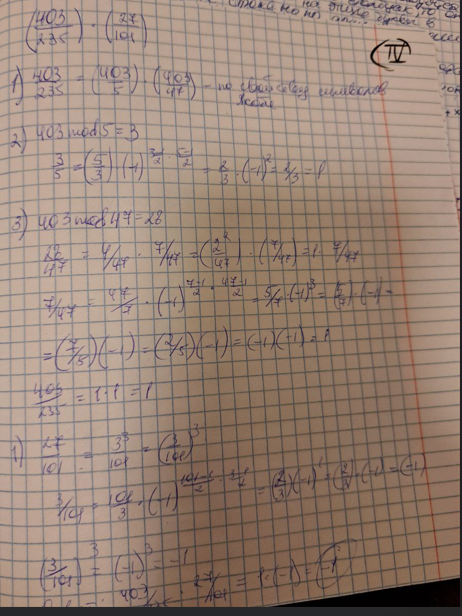
2024

Лабораторная работа №5

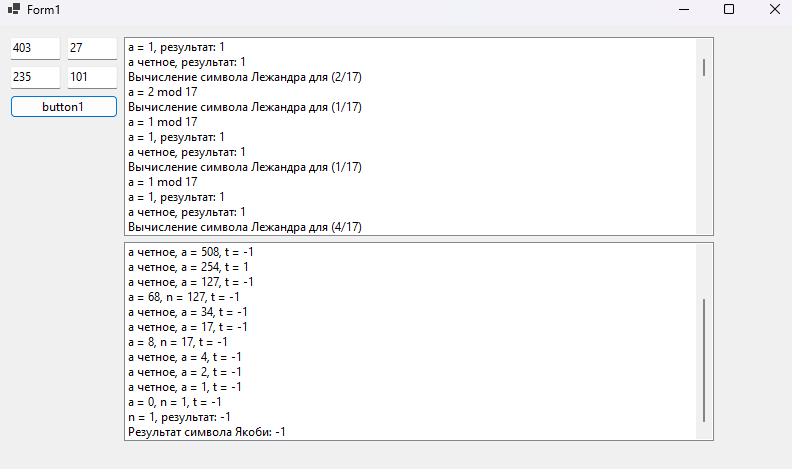
«Символы Лежандра и Якоби.»

Вариант №20

**Ручная реализация.**



**Входные данные.**



**Листинг программы.**

namespace ЯкобиЛежандр

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

// Очистка ListBox перед новым вычислением

listBox1.Items.Clear();

listBox2.Items.Clear();

// Получение значений из TextBox

if (!int.TryParse(textBox1.Text, out int a1) || !int.TryParse(textBox2.Text, out int b1) ||

!int.TryParse(textBox3.Text, out int a2) || !int.TryParse(textBox4.Text, out int b2))

{

MessageBox.Show("Введите целые числа в поля.");

return;

}

// Проверка на положительность знаменателей

if (b1 <= 0 || b2 <= 0)

{

MessageBox.Show("Знаменатели должны быть положительными числами.");

return;

}

// Вычисление произведения дробей

int numerator = a1 \* a2;

int denominator = b1 \* b2;

// Проверка на нечетность знаменателя для символа Лежандра

if (denominator % 2 == 0)

{

MessageBox.Show("Знаменатель для символа Лежандра должен быть нечетным.");

return;

}

// Вычисление символа Лежандра

try

{

int legendreResult = LegendreSymbol(numerator, denominator);

listBox1.Items.Add($"Результат символа Лежандра: {legendreResult}");

}

catch (ArgumentException ex)

{

listBox1.Items.Add(ex.Message);

}

// Вычисление символа Якоби

try

{

int jacobiResult = JacobiSymbol(numerator, denominator);

listBox2.Items.Add($"Результат символа Якоби: {jacobiResult}");

}

catch (ArgumentException ex)

{

listBox2.Items.Add(ex.Message);

}

}

private int LegendreSymbol(int a, int p)

{

listBox1.Items.Add($"Вычисление символа Лежандра для ({a}/{p})");

if (p <= 0 || p % 2 == 0)

throw new ArgumentException("p должно быть положительным нечетным числом.");

a = a % p;

listBox1.Items.Add($"a = {a} mod {p}");

if (a == 0)

{

listBox1.Items.Add("a = 0, результат: 0");

return 0;

}

if (a == 1)

{

listBox1.Items.Add("a = 1, результат: 1");

return 1;

}

if (a < 0)

{

listBox1.Items.Add($"a < 0, результат: {(int)Math.Pow(-1, (p - 1) / 2) \* LegendreSymbol(-a, p)}");

return (int)Math.Pow(-1, (p - 1) / 2) \* LegendreSymbol(-a, p);

}

if (a % 2 == 0)

{

listBox1.Items.Add($"a четное, результат: {LegendreSymbol(a / 2, p) \* (int)Math.Pow(-1, (p \* p - 1) / 8)}");

return LegendreSymbol(a / 2, p) \* (int)Math.Pow(-1, (p \* p - 1) / 8);

}

int result = LegendreSymbol(p % a, a) \* (int)Math.Pow(-1, (a - 1) \* (p - 1) / 4);

listBox1.Items.Add($"a нечетное, результат: {result}");

return result;

}

private int JacobiSymbol(int a, int n)

{

listBox2.Items.Add($"Вычисление символа Якоби для ({a}/{n})");

if (n <= 0 || n % 2 == 0)

throw new ArgumentException("n должно быть положительным нечетным числом.");

a = a % n;

listBox2.Items.Add($"a = {a} mod {n}");

int t = 1;

while (a != 0)

{

while (a % 2 == 0)

{

a /= 2;

int r = n % 8;

if (r == 3 || r == 5)

t = -t;

listBox2.Items.Add($"a четное, a = {a}, t = {t}");

}

int temp = a;

a = n;

n = temp;

if (a % 4 == 3 && n % 4 == 3)

t = -t;

a = a % n;

listBox2.Items.Add($"a = {a}, n = {n}, t = {t}");

}

if (n == 1)

{

listBox2.Items.Add($"n = 1, результат: {t}");

return t;

}

listBox2.Items.Add("n != 1, результат: 0");

return 0;

}

}

}

**Вывод по работе:**

В ходе выполнения лабораторной работы были изучены лекционный материал и способы поиска символа Лежандра и символа Якоби.