

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 №7

по дисциплине: «Теоретико-числовые методы в криптографии»

на тему: «Непрерывные дроби.»

Выполнил обучающийся гр. ВКБ42

Михайлов Александр Сергеевич

          Проверила:

Ст. пр. Артамонова Е.А.

Ростов-на-Дону

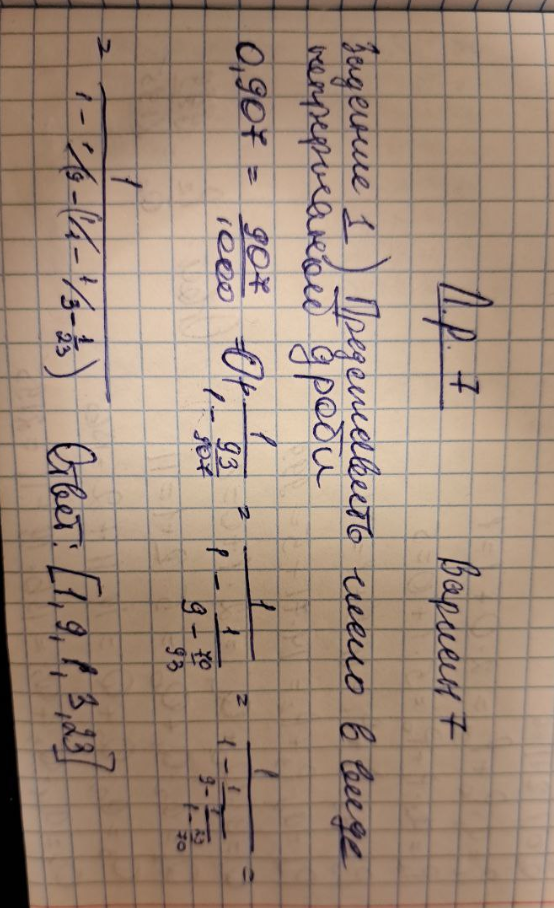
2024

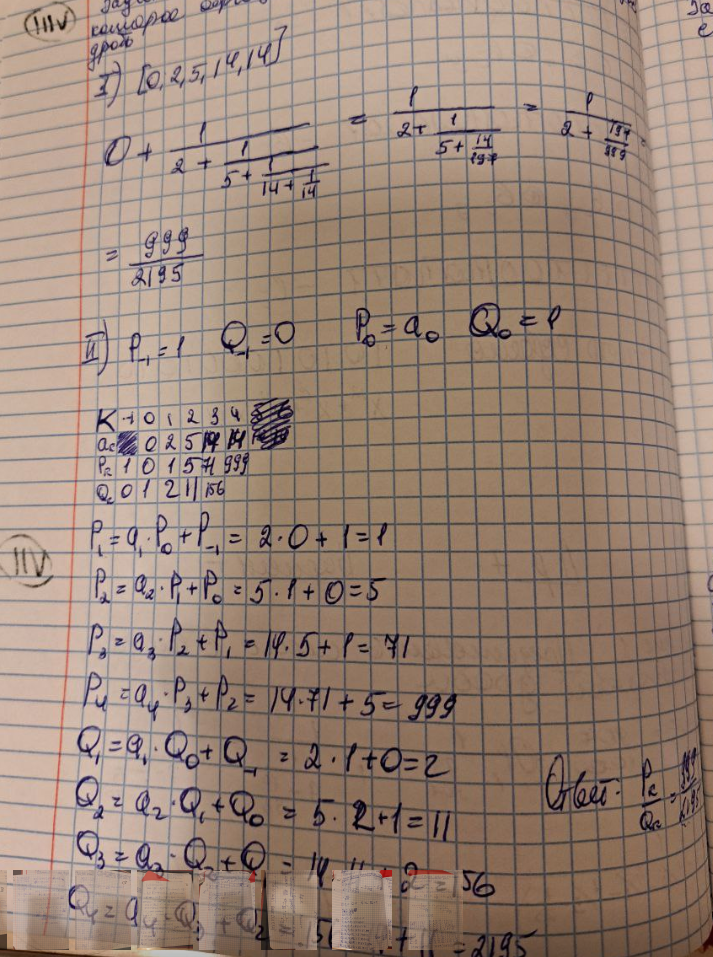
Лабораторная работа №7

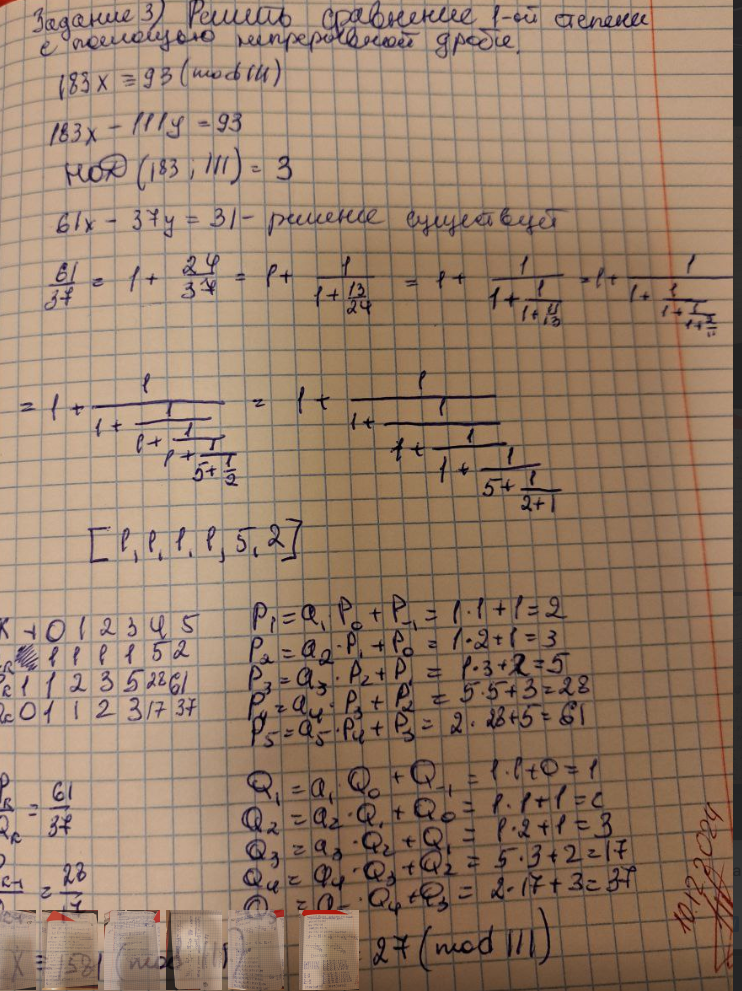
«Непрерывные дроби.»

Вариант №20

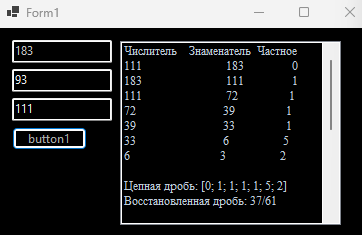
**Ручная реализация.**

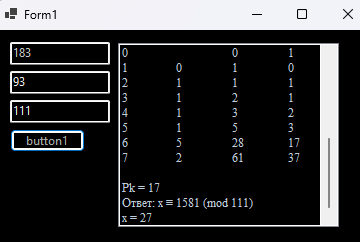






**Входные данные.**

****

****

**Листинг программы.**

namespace НеправильнаДробь

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

}

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

{

int otvet = 0;

listBox1.Items.Clear();

// Проверка входных данных

if (!int.TryParse(textBox1.Text, out int a) ||

!int.TryParse(textBox2.Text, out int b) ||

!int.TryParse(textBox3.Text, out int m))

{

listBox1.Items.Add("Введите корректные значения a, b и m.");

return;

}

listBox1.Items.Add($"Решаем сравнение: {a}x ≡ {b} (mod {m})");

listBox1.Items.Add($"Преобразуем: x ≡ {b}/{a} (mod {m})");

var continuedFraction = GetContinuedFraction(m, a, listBox1);

listBox1.Items.Add(new string(' ', 20));

listBox1.Items.Add($"Цепная дробь: [{string.Join("; ", continuedFraction)}]");

var (numerator, denominator) = ComputeFractionFromContinuedFraction(continuedFraction);

listBox1.Items.Add($"Восстановленная дробь: {numerator}/{denominator}");

listBox1.Items.Add(new string(' ', 20));

listBox1.Items.Add($"{"k",-3} {" ",-12} {"a",-16} {"P",-16} {"Q",-15}");

otvet = ComputeTable(continuedFraction);

listBox1.Items.Add(new string(' ', 20));

listBox1.Items.Add($"Pk = {otvet}");

otvet = otvet \* b;

listBox1.Items.Add($"Ответ: x ≡ {otvet} (mod {m})");

int remainder = otvet % m;

if (remainder < 0)

{

remainder += m;

}

listBox1.Items.Add($"x = {remainder}");

}

private List<int> GetContinuedFraction(int numerator, int denominator, ListBox listBox)

{

var fractions = new List<int>();

listBox.Items.Clear();

// Заголовок таблицы

listBox.Items.Add($"{"Числитель",-12} {"Знаменатель",-12} {"Частное",-10}");

while (denominator != 0)

{

int quotient = numerator / denominator;

int remainder = numerator % denominator;

// Вывод строки таблицы

listBox.Items.Add($"{numerator,-30} {denominator,-18} {quotient,-20}");

fractions.Add(quotient);

numerator = denominator;

denominator = remainder;

}

return fractions;

}

private (int numerator, int denominator) ComputeFractionFromContinuedFraction(List<int> coefficients)

{

int n = coefficients[coefficients.Count - 1];

int d = 1;

for (int k = coefficients.Count - 1; k > 0; k--)

{

int temp = d;

d = n;

n = coefficients[k - 1] \* n + temp;

}

return (n, d);

}

private int ComputeTable(List<int> coefficients)

{

int k = coefficients.Count;

int[] Q = new int[k + 1];

int[] P = new int[k + 1];

Q[0] = 0;

Q[1] = 1;

P[0] = 1;

P[1] = coefficients[0];

listBox1.Items.Add($"0\t\t0\t1\t1");

listBox1.Items.Add($"1\t{coefficients[0]}\t1\t{coefficients[0]}");

for (int i = 2; i <= k; i++)

{

P[i] = coefficients[i - 1] \* P[i - 1] + P[i - 2];

Q[i] = coefficients[i - 1] \* Q[i - 1] + Q[i - 2];

listBox1.Items.Add($"{i}\t{coefficients[i - 1]}\t{Q[i]}\t{P[i]}");

}

int result = P[k - 1] \* (int)Math.Pow(-1, k - 1);

return result;

}

}

}

**Вывод по работе:**

В ходе выполнения лабораторной работы были изучены лекционный материал и поиски непрерывных дробей, и решение с помощью непрерывных дробей сравнений первой степени.