**Липецкий государственный технический университет**

Университетский колледж

Кафедра автоматизированных систем управления

Лабораторная работа №10

по «Защита информации»

Реализация схемы электронной цифровой подписи

Студент Щеглов Д.A

Группа T-ИС-20

Руководитель Седых Ю.И

Липецк 2022г.

Оглавление

[Цель работы 3](#_Toc98706477)

[Задание 4](#_Toc98706478)

# **Цель работы**

Ознакомиться со схемами цифровой подписи и получить навыки создания и проверки подлинности электронной цифровой подписи.

# **Задание**

Изучить краткие теоретические сведения о принципах организации схем электронных цифровых подписей. Написать программу, реализующую подписывание и проверку целостности файла с использованием алгоритмов CRC и RSA.

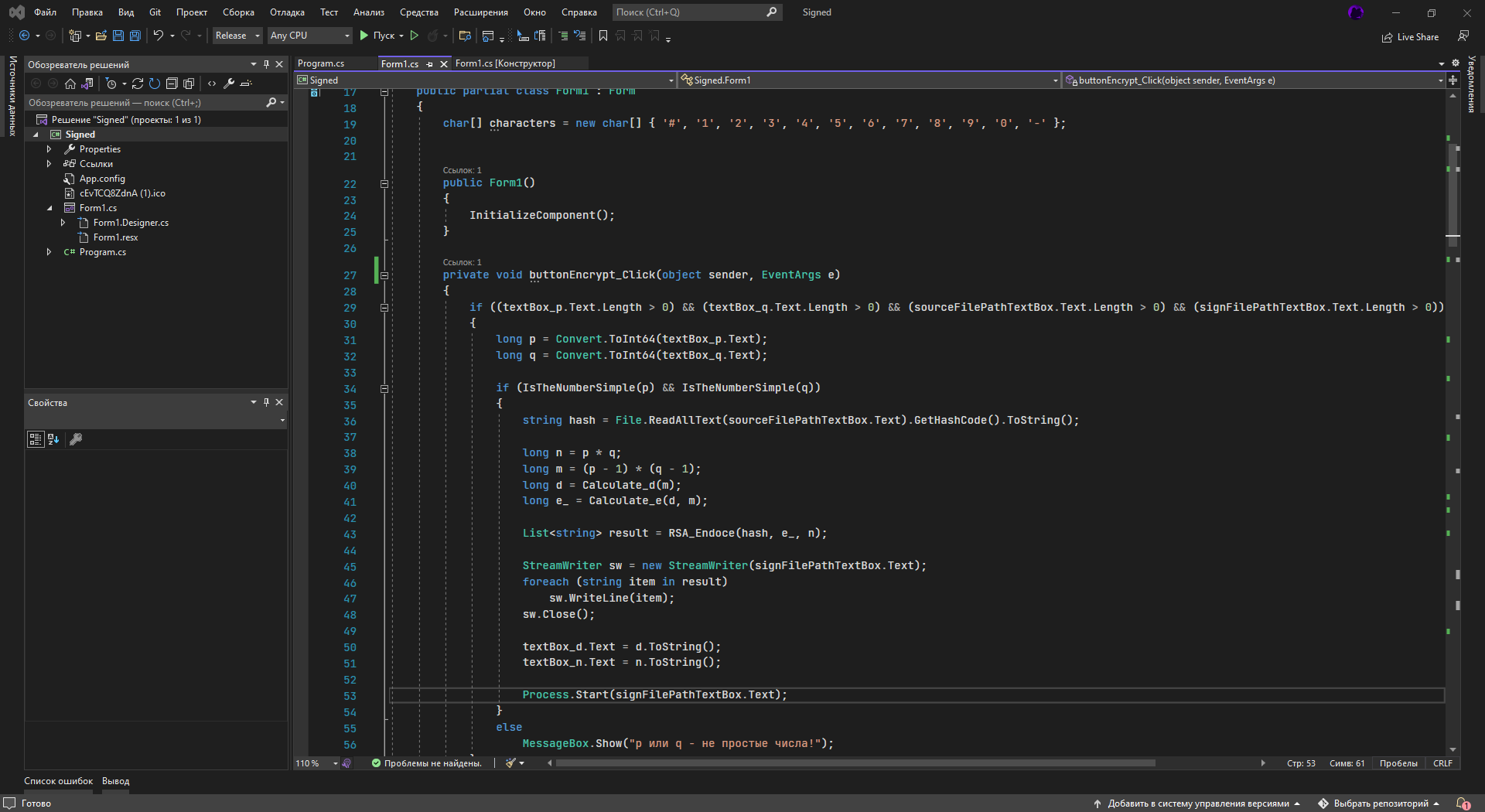


Рис.1

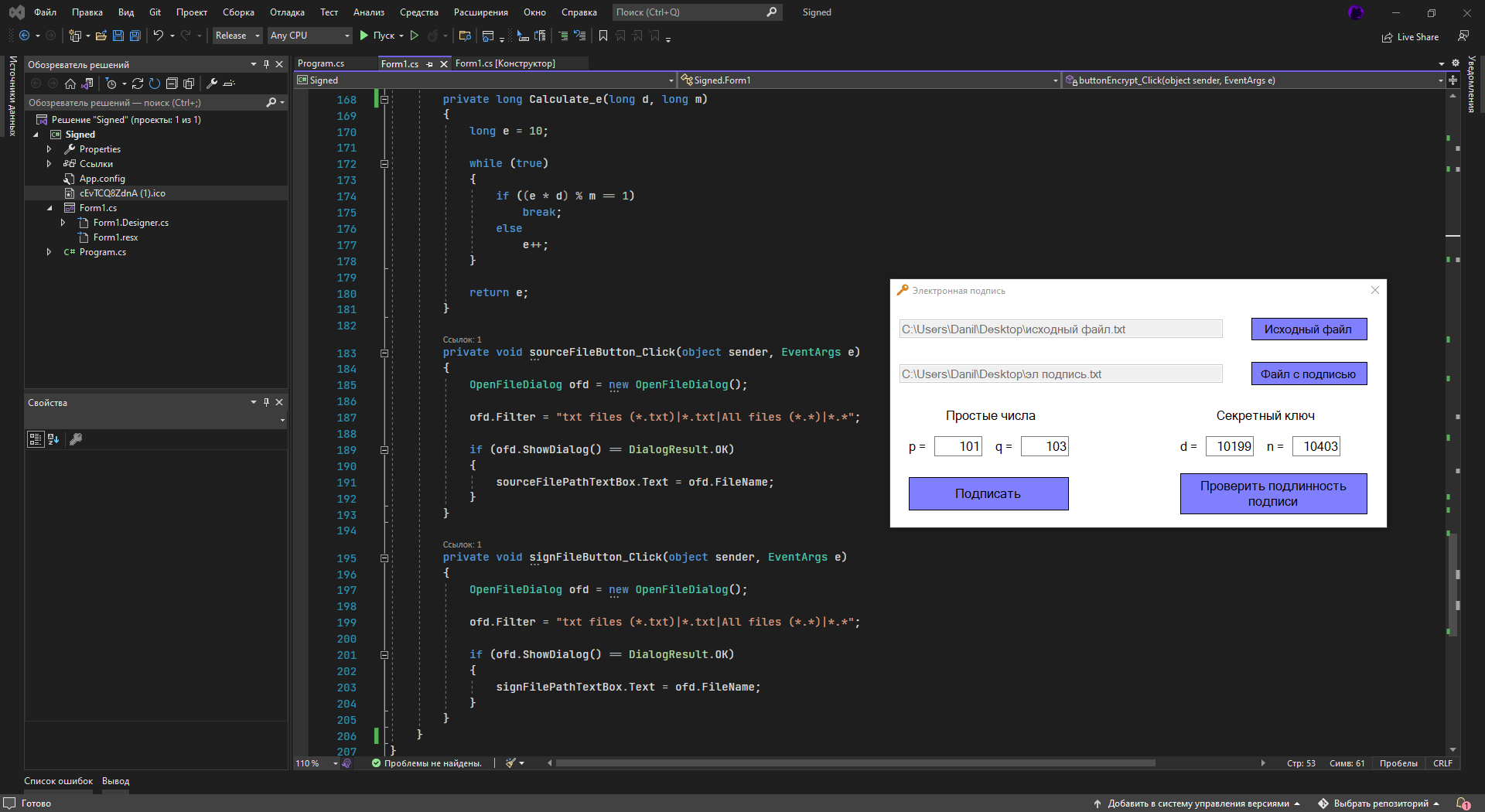


Рис.2

Программный код

using System;

using System.Collections.Generic;

using System.Diagnostics;

using System.IO;

using System.Windows.Forms;

using System.Numerics;

namespace Signed

{

public partial class Form1 : Form

{

char[] characters = new char[] { '#', '1', '2', '3', '4', '5', '6', '7', '8', '9', '0', '-' };

public Form1()

{

InitializeComponent();

}

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

{

if ((textBox\_p.Text.Length > 0) && (textBox\_q.Text.Length > 0) && (sourceFilePathTextBox.Text.Length > 0) && (signFilePathTextBox.Text.Length > 0))

{

long p = Convert.ToInt64(textBox\_p.Text);

long q = Convert.ToInt64(textBox\_q.Text);

if (IsTheNumberSimple(p) && IsTheNumberSimple(q))

{

string hash = File.ReadAllText(sourceFilePathTextBox.Text).GetHashCode().ToString();

long n = p \* q;

long m = (p - 1) \* (q - 1);

long d = Calculate\_d(m);

long e\_ = Calculate\_e(d, m);

List<string> result = RSA\_Endoce(hash, e\_, n);

StreamWriter sw = new StreamWriter(signFilePathTextBox.Text);

foreach (string item in result)

sw.WriteLine(item);

sw.Close();

textBox\_d.Text = d.ToString();

textBox\_n.Text = n.ToString();

Process.Start(signFilePathTextBox.Text);

}

else

MessageBox.Show("p или q - не простые числа!");

}

else

MessageBox.Show("Введите p и q и пути к файлам!");

}

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

{

if ((textBox\_d.Text.Length > 0) && (textBox\_n.Text.Length > 0) && (sourceFilePathTextBox.Text.Length > 0) && (signFilePathTextBox.Text.Length > 0))

{

long d = Convert.ToInt64(textBox\_d.Text);

long n = Convert.ToInt64(textBox\_n.Text);

List<string> input = new List<string>();

StreamReader sr = new StreamReader(signFilePathTextBox.Text);

while (!sr.EndOfStream)

{

input.Add(sr.ReadLine());

}

sr.Close();

string result = RSA\_Dedoce(input, d, n);

string hash = File.ReadAllText(sourceFilePathTextBox.Text).GetHashCode().ToString();

if (result.Equals(hash))

MessageBox.Show("Файл подлинный. Подпись верна.");

else

MessageBox.Show("Внимание! Файл НЕ подлинный!!!");

}

else

MessageBox.Show("Введите секретный ключ и пути к файлам!");

}

private bool IsTheNumberSimple(long n)

{

if (n < 2)

return false;

if (n == 2)

return true;

for (long i = 2; i < n; i++)

if (n % i == 0)

return false;

return true;

}

private List<string> RSA\_Endoce(string s, long e, long n)

{

List<string> result = new List<string>();

BigInteger bi;

for (int i = 0; i < s.Length; i++)

{

int index = Array.IndexOf(characters, s[i]);

bi = new BigInteger(index);

bi = BigInteger.Pow(bi, (int)e);

BigInteger n\_ = new BigInteger((int)n);

bi = bi % n\_;

result.Add(bi.ToString());

}

return result;

}

private string RSA\_Dedoce(List<string> input, long d, long n)

{

string result = "";

BigInteger bi;

foreach (string item in input)

{

bi = new BigInteger(Convert.ToDouble(item));

bi = BigInteger.Pow(bi, (int)d);

BigInteger n\_ = new BigInteger((int)n);

bi = bi % n\_;

int index = Convert.ToInt32(bi.ToString());

result += characters[index].ToString();

}

return result;

}

private long Calculate\_d(long m)

{

long d = m - 1;

for (long i = 2; i <= m; i++)

if ((m % i == 0) && (d % i == 0))

{

d--;

i = 1;

}

return d;

}

private long Calculate\_e(long d, long m)

{

long e = 10;

while (true)

{

if ((e \* d) % m == 1)

break;

else

e++;

}

return e;

}

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

{

OpenFileDialog ofd = new OpenFileDialog();

ofd.Filter = "txt files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

if (ofd.ShowDialog() == DialogResult.OK)

{

sourceFilePathTextBox.Text = ofd.FileName;

}

}

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

{

OpenFileDialog ofd = new OpenFileDialog();

ofd.Filter = "txt files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

if (ofd.ShowDialog() == DialogResult.OK)

{

signFilePathTextBox.Text = ofd.FileName;

}

}

}

}