Федеральное государственное образовательное бюджетное учреждение

высшего образования «Финансовый университет при Правительстве Российской Федерации»

**Колледж информатики и программирования**

**Специальность 09.02.03 Программирование в компьютерных системах**

**ПМ.03 Участие в интеграции программных модулей**

**МДК.03.01. Технология разработки программного обеспечения**

ПРАКТИЧЕСКАЯ РАБОТА №4

Выполнил:

студент группы: 3ПКС-115

Черников А. В.

Проверила:

Морозова М. В.

Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва 2018

**Практическая работа №4**

**Условие задачи:**

Задание:

Написать программу с графическим интерфейсом которая будет находить наибольший общий делитель для 2-5 чисел. Сделать справку для данной программы.

**Код программы:**

Модуль MainWindow.xaml.cs:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using System.Diagnostics;

namespace GreatestCommonDivisor{

/// <summary>

/// Interaction logic for MainWindow.xaml

/// </summary>

public partial class MainWindow : Window{

public MainWindow(){

InitializeComponent();

}

/// <summary>

/// Do the GCD calculations

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void FindGCD\_Click(object sender, RoutedEventArgs e){

int firstNumber;

int secondNumber;

int thirdNumber;

int fourthNumber;

int fifthNumber;

if (!GetPostiveIntegerFromTextBox(integer1, out firstNumber)) return;

if (!GetPostiveIntegerFromTextBox(integer2, out secondNumber)) return;

if (!GetPostiveIntegerFromTextBox(integer3, out thirdNumber)) return;

if (!GetPostiveIntegerFromTextBox(integer4, out fourthNumber)) return;

if (!GetPostiveIntegerFromTextBox(integer5, out fifthNumber)) return;

// Display results

if (sender == findGCD) // Euclid for two integers{

resultEuclid.Content = string.Format("Euclid: {0}", GCDAlgorithms.FindGCDEuclid(firstNumber, secondNumber));

}

if (sender == findGCD3) // Euclid for three integers{

int[] param = { firstNumber, secondNumber, thirdNumber };

resultEuclid.Content = string.Format("Euclid: {0}", GCDAlgorithms.FindGCDEuclid(param));

}

if (sender == findGCD4) // Euclid for four integers{

int[] param = { firstNumber, secondNumber, thirdNumber, fourthNumber };

resultEuclid.Content = string.Format("Euclid: {0}", GCDAlgorithms.FindGCDEuclid(param));

}

if (sender == findGCD5) // Euclid for five integers{

int[] param = { firstNumber, secondNumber, thirdNumber, fourthNumber, fifthNumber };

resultEuclid.Content = string.Format("Euclid: {0}", GCDAlgorithms.FindGCDEuclid(param));

}

if (sender == findStein2) // Stein for two integers{

resultStein.Content = string.Format("Stein: {0}", GCDAlgorithms.FindGCDStein(firstNumber, secondNumber));

}

}

/// <summary>

/// Read a postive integer from a TextBox

/// Displays a message box with the data if the text can't be parsed.

/// </summary>

/// <param name="textBox">TextBox to read</param>

/// <param name="i">Postive integer (out parameter)</param>

/// <returns>True if success, false otherwise</returns>

private bool GetPostiveIntegerFromTextBox(TextBox textBox, out int i){

i = -1;

if (int.TryParse(textBox.Text, out i))

{

if (i >= 0) return true;

}

MessageBox.Show("Not a positive integer value: " + textBox.Text);

return false;

}

private void Button\_Click\_1(object sender, RoutedEventArgs e){

string commandText = "C:\\Users\\kbstudentint\\Desktop\\12.chm";

var proc = new System.Diagnostics.Process();

proc.StartInfo.FileName = commandText;

proc.StartInfo.UseShellExecute = true;

proc.Start();

}

private void Window\_KeyDown\_1(object sender, KeyEventArgs e){

KeyConverter converter = new KeyConverter();

string key = converter.ConvertToString(e.Key);

if (key == "F1") {

string commandText = "C:\\Users\\kbstudentint\\Desktop\\12.chm";

var proc = new System.Diagnostics.Process();

proc.StartInfo.FileName = commandText;

proc.StartInfo.UseShellExecute = true;

proc.Start();

}

}

}

}

Модуль GCDAlgoritms:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Diagnostics;

namespace GreatestCommonDivisor

{

static class GCDAlgorithms

{

static public int FindGCDEuclid( int a, int b)

{

int r=0;

while (a > 0 && b > 0)

if (a > b)

a %= b;

else

b %= a;

r = a + b;

return r;

}

static public int gcd(int a, int b)

{

if (a == 0)

return b;

if (b == 0)

return a;

int k;

for (k = 0; ((a | b) & 1) == 0; ++k)

{

a >>= 1;

b >>= 1;

}

while ((a & 1) == 0)

a >>= 1;

do

{

while ((b & 1) == 0)

b >>= 1;

if (a > b)

{

int tmp;

tmp = a;

a = b;

b = tmp;

}

b = (b - a);

} while (b != 0);

return a << k;

}

static public int FindGCDEuclid(int a, int b,int c)

{

while (b != 0)

b = a % (a = b);

return FindGCDEuclid(a, c);

}

static public int FindGCDEuclid(int a, int b, int c,int d)

{

while (b != 0)

b = a % (a = b);

return FindGCDEuclid(a, c, d);

}

static public int FindGCDEuclid(int a, int b, int c,int d,int e)

{

while (b != 0)

b = a % (a = b);

return FindGCDEuclid(a, c, d, e);

}}}

**Результаты:**

Результат задачи №1 представлен на рисунке 1, 2, 3

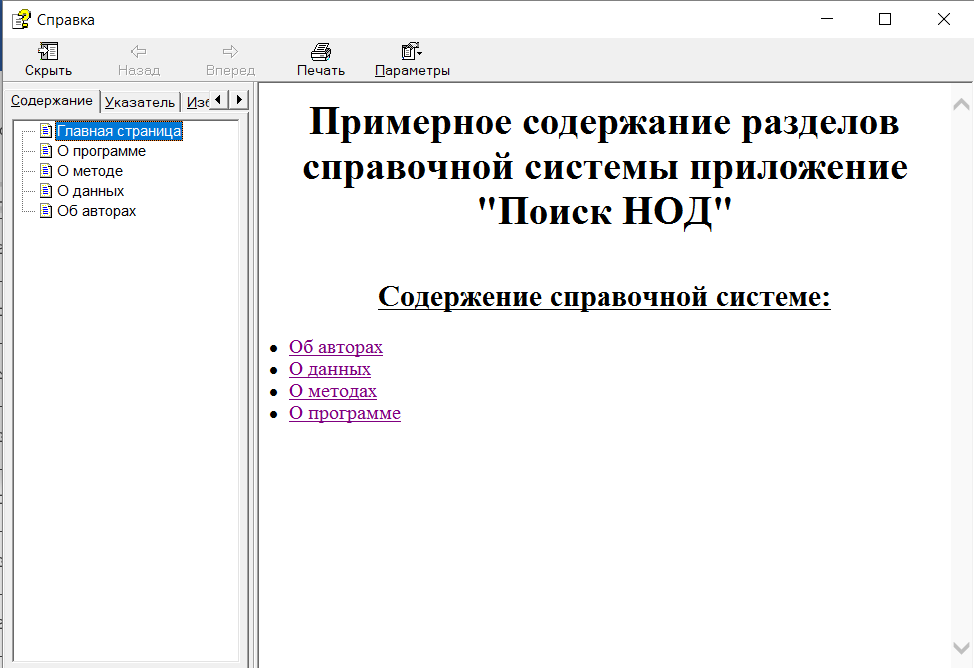


Рисунок 1

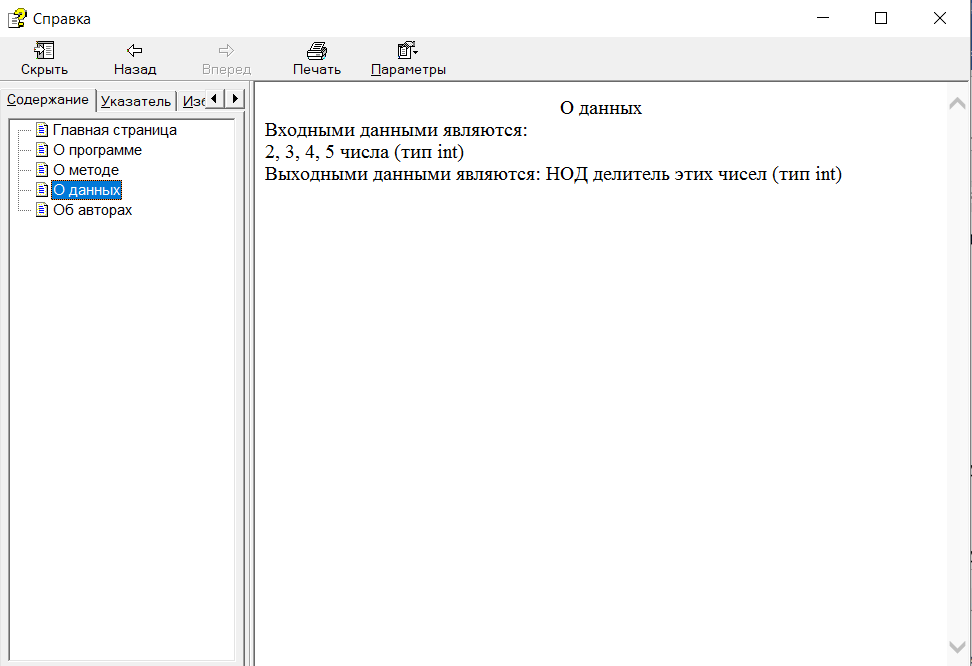


Рисунок 2

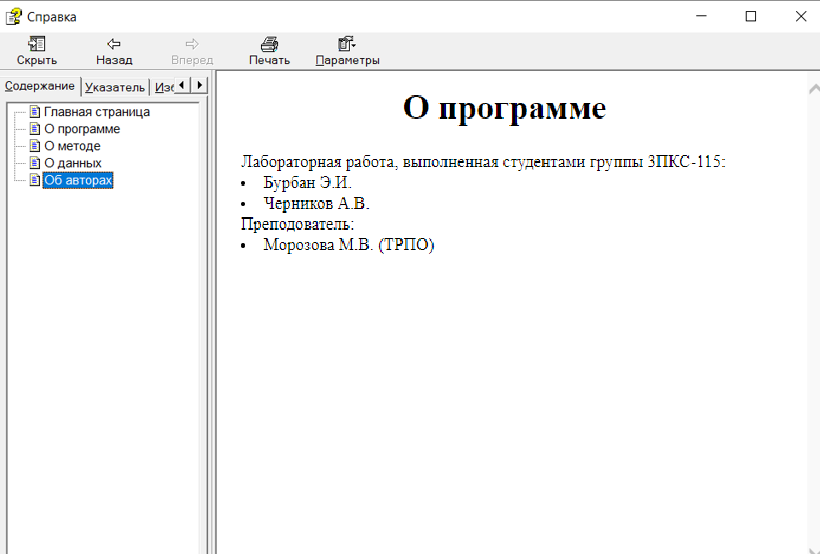


Рисунок 3