**Министерство высшего образования и науки Российской Федерации**

Федеральное государственное бюджетное образовательное учреждение Высшего образования

«**Владимирский государственный университет**

**имени Александра Григорьевича и Николая Григорьевича Столетовых»**

**(ВлГУ)**

Колледж информационных технологий и предпринимательства (КИТП)

**Кафедра физики и прикладной математики (ФиПМ)**

Задание 4

Тема: «**Реализация рекурсивных методов**»

Вариант 2, 12

Выполнил студент группы ПКсп-120

Башилов М.А.

Принял(а) Кабанова М. Ю.

Владимир 2022

Постановка задачи

Требуется реализовать консольное приложение и приложение с графическим интерфейсом на языке программирования C#, выполнив следующие задания:

1. Составить программу, которая содержит рекурсивный метод, который вычисляет n-й член последовательности.
2. Составить программу, которая запрашивает натуральное четное число n и выводит на экран картинку, согласно образцу с помощью рекурсивного метода.

Практическая часть

Листинг первого консольного приложения

using System;

namespace Ex1\_Console

{

internal class Program

{

static void Main(string[] args)

{

int n = 2;

while (true)

{

try

{

Console.Write("Введите целое число для вычисления n-го члена последовательности: ");

n = int.Parse(Console.ReadLine());

if (n < 1) throw new Exception("Число должно быть больше или равно 1!");

break;

}

catch (FormatException)

{

Console.ForegroundColor = ConsoleColor.Red;

Console.Write("Ошибка: ");

Console.ResetColor();

Console.WriteLine("Ожидается ввод целого числа!");

}

catch (Exception ex)

{

Console.ForegroundColor = ConsoleColor.Red;

Console.Write("Ошибка: ");

Console.ResetColor();

Console.WriteLine(ex.Message);

}

}

Console.WriteLine($"\nРезультат вычисления последовательности: {Calculate(5.0, n)}");

}

static double Calculate(double b, int n)

{

if (n <= 1) return b;

return Calculate(b / ((n \* n) + n + 1), n - 1);

}

}

}

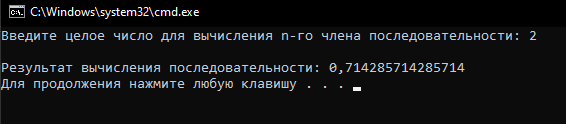


Рисунок 1 – Результат работы программы первого задания

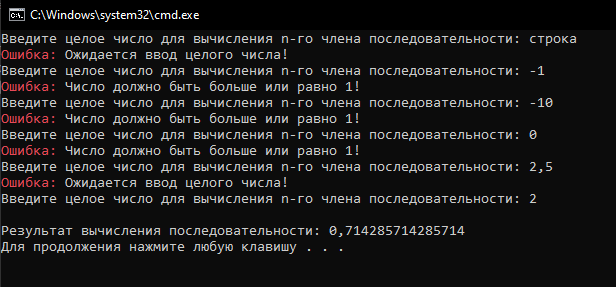


Рисунок 2 – Результат работы программы при вводе некорректных данных

Листинг второго консольного приложения

using System;

namespace Ex2\_Console

{

internal class Program

{

static void Main(string[] args)

{

int n = 0;

while (true)

{

try

{

Console.Write("Введите положительное четное целое число: ");

n = int.Parse(Console.ReadLine());

if (n < 1) throw new Exception("Число должно быть больше или равно 1!");

if (n % 2 != 0) throw new Exception("Число должно быть четным!");

break;

}

catch (FormatException)

{

Console.ForegroundColor = ConsoleColor.Red;

Console.Write("Ошибка: ");

Console.ResetColor();

Console.WriteLine("Ожидается ввод целого числа!");

}

catch (Exception ex)

{

Console.ForegroundColor = ConsoleColor.Red;

Console.Write("Ошибка: ");

Console.ResetColor();

Console.WriteLine(ex.Message);

}

}

ShowPicture(0, n);

}

static void ShowPicture(int spaces, int n)

{

if (spaces == n) return;

int space = spaces;

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

Console.Write(" ");

for (int i = space; i < n; i++)

Console.Write("\*");

Console.WriteLine();

ShowPicture(spaces + 1, n);

}

}

}

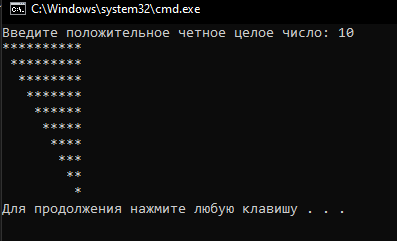


Рисунок 3 – Результат работы программы второго задания

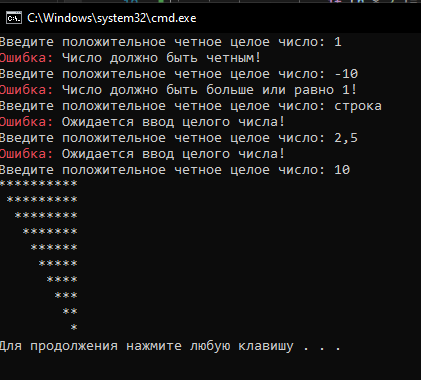


Рисунок 4 – Результат работы программы при некорректных значениях

Листинг графического приложения

using System;

using System.Windows.Forms;

namespace Ex1\_Forms

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

int n = 0;

try

{

Console.Write("Введите целое число для вычисления n-го члена последовательности: ");

n = int.Parse(NTextBox.Text);

if (n < 1) throw new Exception("Число должно быть больше или равно 1!");

}

catch (FormatException)

{

MessageBox.Show("Ожидается ввод целого числа!", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

MessageBox.Show($"Результат вычисления последовательности: {Calculate(5.0, n)}", "Результат", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private double Calculate(double b, int n)

{

if (n <= 1) return b;

return Calculate(b / ((n \* n) + n + 1), n - 1);

}

}

}

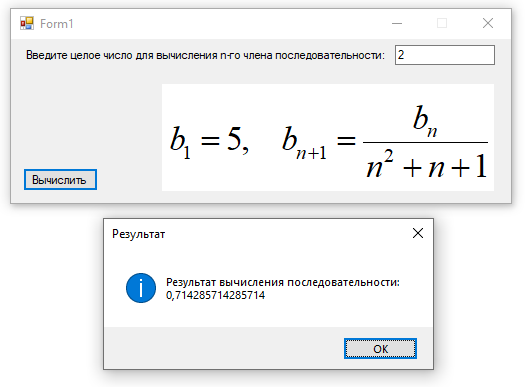


Рисунок 5 – Графический интерфейс программы первого задания

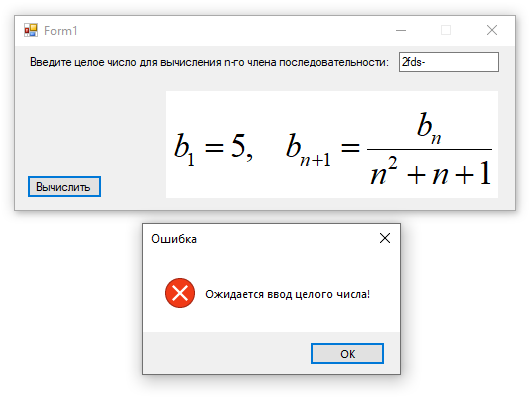


Рисунок 6 – Ввод некорректных данных в программу

Листинг второго графического приложения

using System;

using System.Windows.Forms;

namespace Ex2\_Forms

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

int n = 10;

try

{

n = int.Parse(NTextBox.Text);

if (n < 1) throw new Exception("Число должно быть больше или равно 1!");

if (n % 2 != 0) throw new Exception("Число должно быть четным!");

}

catch (FormatException)

{

MessageBox.Show("Ожидается ввод целого положительного числа!", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

ResultTextBox.Clear();

ShowPicture(0, n);

}

private void ShowPicture(int spaces, int n)

{

if (spaces == n) return;

int space = spaces;

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

ResultTextBox.Text += ' ';

for (int i = space; i < n; i++)

ResultTextBox.Text += '\*';

ResultTextBox.Text += '\n';

ShowPicture(spaces + 1, n);

}

}

}

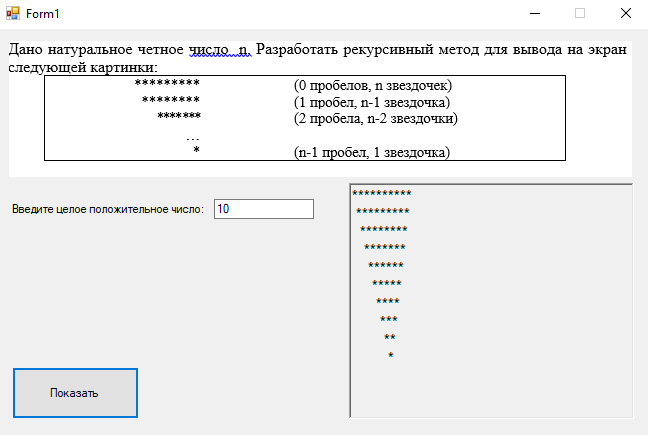


Рисунок 7 – Графический интерфейс программы второго задания

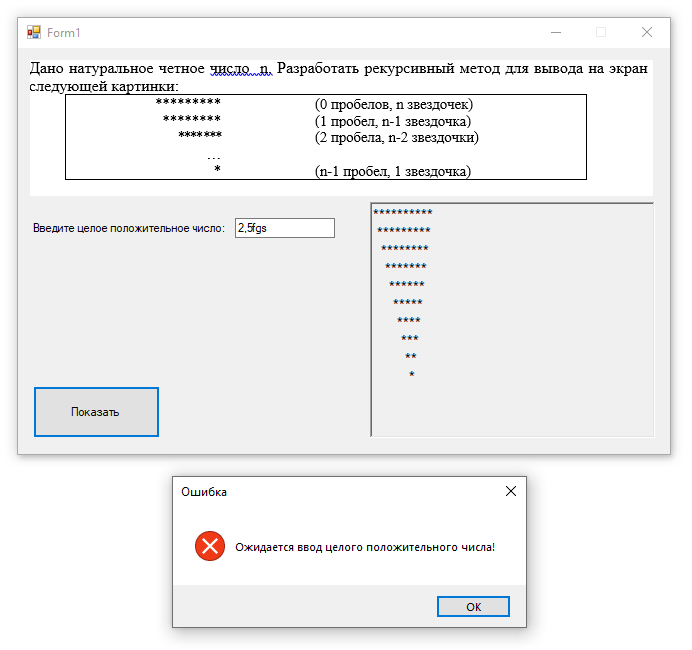
****

Рисунок 8. Графический интерфейс программы второго задания при некорректных данных

Ссылка на GitHub

Все работы будут храниться в следующих репозиториях:

1. Репозиторий со всеми консольными реализациями заданий:

https://github.com/LOZH-dot/SP4\_Bashilov

1. Репозиторий со всеми реализациями заданий с графическим интерфейсом:

https://github.com/LOZH-dot/SP4\_Bashilov