

**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**

**(РУТ (МИИТ)**

**ИНСТИТУТ ТРАНСПОРТНОЙ ТЕХНИКИ И СИСТЕМ УПРАВЛЕНИЯ**

**Лабораторная работа №3**

**по дисциплине** «**Методы программирования**»

«Автоматизация компоновки ГПИ. Функции(варианты)»

**Выполнил:** ст. гр. ТКИ-341

Соколов И.Д.

Ковров А.И.

**Проверил:** к.т.н. Сафронов А.И.

**Москва – 2024 г**



1. **Цель работы (одна из возможных формулировок)**: «Закрепить навыки разработки визуального пользовательского интерфейса, освоить работу с текстовыми файлами в среде *Microsoft Visual Studio*, научиться взаимно увязывать одни элементы управления с другими, получить представление о перерисовке и перемасштабировании».
2. **Основное задание на работу**: «В интегрированной среде разработки *Visual Studio* разработать программу в режиме *Windows Forms Application* на языке *Visual C#*, представляющую собой пользовательский интерфейс, содержащий главное меню, позволяющее:
3. Начать работу с приложением.
4. Прервать работу приложения.
5. Предоставить пользователю справочную информацию о работе с приложением.

Сама программа должна реализовывать вывод в списки значений аргумента и соответствующих им значений функций. Список функций должен обязательно содержать следующие пункты:

1. Логарифм по основанию 2,
2. Тангенс,
3. Возведение в квадрат,
4. Косинус,
5. Натуральный логарифм,
6. Арккосинус,
7. Извлечение корня,
8. Арктангенс,
9. Синус,
10. Десятичный логарифм,
11. Арксинус.

# **Детализированное текстовое описание ситуации**

Пользователь открывает программу, запускается пустая начальная форма, за ней запускается вторая форма заполненная по заданию. В данной форме интерфейс закреплен за левым верхним углом, при изменении размера формы интерфейс сохраняет свой размер и сохраняет закрепление за левым верхним углом. Минимальный размер формы установлен с помощью кода(450\*200). В интерфейсе второй формы есть кнопка help, которая вызывает окно помощи. Создан объект combobox1 который вызывает окно выбора функции по заданию. Создан объект textbox1 который вызывает окно, где пользователь может ввести диапазон для x. После нажатия кнопки start в listbox1 появляются значения x, в listbox2 появляются значения x после применения функции. При выборе в listbox1 какого-либо x в listbox2 выбирается соответствующее ему значение функции и наоборот.

1. **Сеть Петри – схема ситуации**
   1. **Полная**

Изображение выглядит как диаграмма, зарисовка, рисунок, круг

Автоматически созданное описание

Рисунок 1 – Полная сеть Петри

1. **Тестовые примеры**

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 2 – Выбор функции

Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Автоматически созданное описание

Рисунок 3 – Выбор диапазона x

Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Автоматически созданное описание

Рисунок 4 – Форма help

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 5 – Вывод x и x ^ 2

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 6 – Вывод x и sin(x)

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 7 – Выделение x и вывод соответствующей func(x)

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Автоматически созданное описание

Рисунок 8 – Изменение размера формы

Изображение выглядит как текст, программное обеспечение, снимок экрана, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 9 – Начальная пустая форма

1. **Диаграмма классов**

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 10 – Диаграмма классов

1. **Листинг**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Lab3

{

public partial class main : Form

{

public main()

{

InitializeComponent();

}

string[] mas = { "Ничего не выбрано","Arctg(x)", "Arccos(x)", "Arcsin(x) ", "tg(x) ", "sin(x) ", "x/cos(x) ", "x^2 ", "cos(x) ", "ln(x) ", "log2(x) ", "x^(1/2) " };

int index\_combobox = 0;

bool flag\_error = false;

ComboBox combobox1;

Button newButton1;

Button newButton2;

ListBox listbox1;

ListBox listbox2;

TextBox textbox1;

double[] abr = { -5, 5, 0.1 };

int index\_x = 0;

int index\_y = 0;

private void Create\_all(object sender, EventArgs e)

{

newButton1 = new Button();

this.Controls.Add(newButton1);

newButton1.Click += new EventHandler(newButton1\_Click);

newButton1.Text = "Start";

newButton1.Location = new Point(10, 10);

newButton1.Size = new Size(60, 30);

newButton1.ForeColor = Color.Red;

newButton2 = new Button();

this.Controls.Add(newButton2);

newButton2.Click += new EventHandler(newButton2\_Click);

newButton2.Text = "Help";

newButton2.Location = new Point(180, 10);

newButton2.Size = new Size(60, 30);

newButton2.ForeColor = Color.Red;

combobox1 = new ComboBox();

this.Controls.Add(combobox1);

combobox1.Location = new Point(10, 110);

foreach(string str in mas)

{

combobox1.Items.Add(str);

}

combobox1.SelectedIndex = 0;

combobox1.SelectedIndexChanged += new EventHandler(combobox1\_SelectedIndexChanged);

listbox1 = new ListBox();

this.Controls.Add(listbox1);

listbox1.Location = new Point(150, 50);

listbox1.SelectedIndexChanged += new EventHandler(listBox1\_SelectedIndexChanged);

listbox2 = new ListBox();

this.Controls.Add(listbox2);

listbox2.Location = new Point(300, 50);

listbox2.SelectedIndexChanged += new EventHandler(listBox2\_SelectedIndexChanged);

textbox1 = new TextBox();

this.Controls.Add(textbox1);

textbox1.Location = new Point(10, 60);

textbox1.Text = abr[0].ToString() + ";" + abr[1].ToString() + ";" + abr[2].ToString();

textbox1.TextChanged += new EventHandler(textbox1\_TextChanged);

newButton1.Anchor = AnchorStyles.Left | AnchorStyles.Top;

newButton2.Anchor = AnchorStyles.Left | AnchorStyles.Top;

textbox1.Anchor = AnchorStyles.Left | AnchorStyles.Top;

listbox1.Anchor = AnchorStyles.Left | AnchorStyles.Top;

listbox2.Anchor = AnchorStyles.Left | AnchorStyles.Top;

combobox1.Anchor = AnchorStyles.Left | AnchorStyles.Top;

}

private void listBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

listbox2.SelectedIndex = listbox1.SelectedIndex;

}

private void listBox2\_SelectedIndexChanged(object sender, EventArgs e)

{

listbox1.SelectedIndex = listbox2.SelectedIndex;

}

private void combobox1\_SelectedIndexChanged(object sender, EventArgs e)

{

index\_combobox = combobox1.SelectedIndex;

}

private void textbox1\_TextChanged(object sender, EventArgs e)

{

string[] words = textbox1.Text.Split(';');

abr = new double[3];

int i = 0;

flag\_error = false;

try

{

foreach(string word in words)

{

abr[i] = double.Parse(word);

i++;

}

}

catch(Exception ex)

{

flag\_error = true;

}

}

double[] param = new double[0];

double[] otvet = new double[0];

private void newButton1\_Click(object sender, EventArgs e)

{

if (flag\_error || index\_combobox == 0)

{

MessageBox.Show("wrong Input");

}

else

{

//MessageBox.Show($"{abr[2]}");

listbox1.Items.Clear();

listbox2.Items.Clear();

for (double i = abr[0]; i<= abr[1]; i += abr[2])

{

listbox1.Items.Add(i);

listbox2.Items.Add(formula(i));

}

}

}

private double formula(double x)

{

switch (index\_combobox)

{

case 1:

return Math.Atan(x);

break;

case 2:

return Math.Acos(x);

case 3:

return Math.Asin(x);

break;

case 4:

return Math.Tan(x);

case 5:

return Math.Sin(x);

break;

case 6:

return (x/Math.Cos(x));

case 7:

return x\*x;

break;

case 8:

return Math.Cos(x);

case 9:

return Math.Log(x);

break;

case 10:

return Math.Log10(x)/Math.Log10(2);

case 11:

return Math.Sqrt(x);

}

MessageBox.Show("Error ocured");

return -1;

}

private void newButton2\_Click(object sender, EventArgs e)

{

Form helpp = new Form();

helpp.Text = "Help";

helpp.Width = 400;

helpp.Height = 400;

helpp.BackColor = SystemColors.Highlight;

Label label1 = new Label();

label1.Location = new System.Drawing.Point(300, 100);

helpp.Controls.Add(label1);

label1.Width = 270;

label1.Height = 40;

label1.Text = "This form could help you,\n but it won't(use , for 0,1)";

label1.ForeColor = Color.Red;

label1.Location = new Point(10, 10);

helpp.ShowDialog();

}

private void main\_Load\_1(object sender, EventArgs e)

{

InitializeComponent();

this.BackColor = SystemColors.Highlight;

this.Shown += Create\_all;

this.MinimumSize = new Size(450, 200);

this.Text = "08032024 Task 3 Kovrov Sokolov TKI-341";

}

}

}

1. **Вывод**

Мы закрепили навыки разработки визуального пользовательского интерфейса, освоили работу с текстовыми файлами в среде *Microsoft Visual Studio*, научились взаимно увязывать одни элементы управления с другими, получили представление о перерисовке и перемасштабировании