**22 Технология Windows Form. Работа с формами**

Задание №1. Составить программу вычисления для заданных значений x = 2.444, y = 0.869 \* 10-2, z = -0.13 \* 103 арифметического выражения:



Листинг программы:

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

{

textBox1.Text = 2.444.ToString();

textBox2.Text = (0.869 \* Math.Pow(10, -2)).ToString();

textBox3.Text = (-0.13 \* Math.Pow(10, 3)).ToString();

}

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

{

double constantE = 2.7182;

textBox4.Text = "Laboratory work 22, Task 1, Leonov A.A.";

double x = double.Parse(textBox1.Text);

textBox4.Text += Environment.NewLine + "X = " + x.ToString();

double y = double.Parse(textBox2.Text);

textBox4.Text += Environment.NewLine + "Y = " + y.ToString();

double z = double.Parse(textBox3.Text);

textBox4.Text += Environment.NewLine + "Z = " + z.ToString();

double h = Math.Round((Math.Pow(x, y+1) + Math.Pow(constantE, y-1))/(1 + x \* Math.Abs(y - Math.Tan(z))) \* (1 + Math.Abs(y - x)) + (Math.Pow(Math.Abs(y - x), 2) / 2) - (Math.Pow(Math.Abs(y - x), 3) / 3), 4);

textBox4.Text += Environment.NewLine +

"Result h: " + h.ToString();

}

Таблица 22.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 2.444, 0.00869, -130 | -0.4987 |

Источник: собственная разработка

Анализ результата разработки приложения представлен на рисунке 22.1.

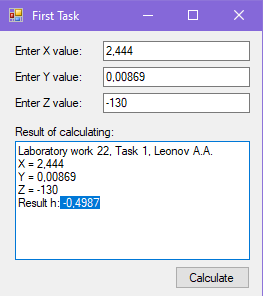


Рисунок 22.1 – Результат работы программы

Источник: собственная разработка

Задание №2. Ввести три числа – x, y, z. Вычислить в соответствии с выражением .

В качестве f(x) использовать по выбору: sh(x), x2, ex. Отредактировать вид формы и текст программы, в соответствии с полученным заданием.

Листинг программы:

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

{

try

{

double x = Convert.ToDouble(textBox1.Text);

double y = Convert.ToDouble(textBox2.Text);

double z = Convert.ToDouble(textBox3.Text);

textBox4.Text = "Laboratory work 22, Task 2, Leonov A.A." + Environment.NewLine;

textBox4.Text += "value X = " + textBox1.Text + Environment.NewLine;

textBox4.Text += "value Y = " + textBox2.Text + Environment.NewLine;

textBox4.Text += "value Z = " + textBox3.Text + Environment.NewLine;

double r = Math.Max(Math.Min(Math.Pow(x, 2), y), z);

textBox4.Text += "result is " + r.ToString() + Environment.NewLine;

}

catch(Exception ex)

{

MessageBox.Show($"An error occured - {ex}");

}

}

Таблица 22.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 4, 6, 9 | 9 |

Источник: собственная разработка

Анализ результата разработки приложения представлен на рисунке 22.2.

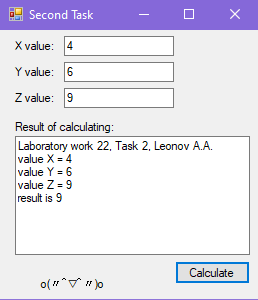


Рисунок 22.2 – Результат работы программы

Источник: собственная разработка

Задание №3. Вычислить и вывести на экран таблицу значений функции y=10-3-|x|-3/2 + ln|x + b| при x, изменяющемся от x0 до xk с шагом dx, a – константа.

Листинг программы:

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

{

textBox5.ScrollBars = ScrollBars.Vertical;

try

{

double x0 = Convert.ToDouble(textBox1.Text);

double xk = Convert.ToDouble(textBox2.Text);

double dx = Convert.ToDouble(textBox3.Text);

double b = Convert.ToDouble(textBox4.Text);

textBox5.Text = "Task 3 done by Leonov A.A. " + Environment.NewLine;

double x = x0;

for (double i = x0; i<=xk; i+=dx)

{

double y = Math.Round(Math.Pow(10, -3) \* Math.Round(Math.Abs(x), 5 / 2) + Math.Log10(Math.Abs(x + b)), 4);

textBox5.Text += "x=" + Convert.ToString(x) + "; y=" + Convert.ToString(y) + Environment.NewLine;

x += dx;

}

}

catch (Exception ex)

{

MessageBox.Show($"An error occured - {ex}");

}

Таблица 22.3 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 1, 6, 0.3, 3 | 0.7662, 0.8854, 0.9025, 0.9503 |

Источник: собственная разработка

Анализ результата разработки приложения представлен на рисунке 22.3.

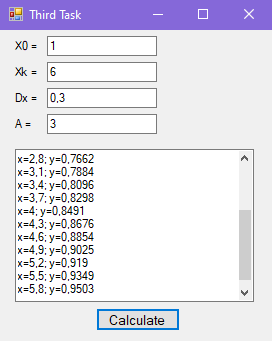


Рисунок 22.3 – Результат работы программы

Источник: собственная разработка

Задание №4. Разработать Windows-приложение, позволяющее пользователю вводить и выводить анкетные данные служащих.

Листинг программы:

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

{

YearDropBox.Items.Clear();

var thisYear = DateTime.Now.Year;

for (int i = 1900; i <= thisYear; i++)

{

YearDropBox.Items.Add(i.ToString());

}

}

private void button4\_Click(object sender, EventArgs e)

{

File.Delete(@"hr\_info.txt");

string path = @"hr\_info.txt";

SaveInfo(this, path);

}

private void button5\_Click(object sender, EventArgs e)

{

ClearFields(this);

}

private void button6\_Click(object sender, EventArgs e)

{

Application.Exit();

}

private static void ClearFields(Control parent)

{

foreach (Control c in parent.Controls)

{

if (c is TextBox)

{

((TextBox)c).Clear();

}

if (c.HasChildren)

{

ClearFields(c);

}

if (c is CheckBox)

{

((CheckBox)c).Checked = false;

}

if (c is RadioButton)

{

((RadioButton)c).Checked = false;

}

if (c is ComboBox)

{

((ComboBox)c).Text = "";

}

}

}

private static void SaveInfo(Control parent, string path)

{

foreach (Control c in parent.Controls)

{

if (c is TextBox txt)

{

File.AppendAllText(path, txt.Name + ": " + txt.Text + "\n");

}

if (c.HasChildren)

{

if (!(c is NumericUpDown))

{

SaveInfo(c, path);

}

}

if (c is CheckBox)

{

if (((CheckBox)c).Checked)

{

File.AppendAllText(path, c.Text + ": " + ((CheckBox)c).Checked.ToString() + "\n");

}

}

if (c is RadioButton)

{

if (((RadioButton)c).Checked)

{

File.AppendAllText(path, c.Text + ": " + ((RadioButton)c).Checked.ToString() + "\n");

}

}

if (c is ComboBox)

{

File.AppendAllText(path, c.Name + ": " + ((ComboBox)c).Text + "\n");

}

if (c is NumericUpDown)

{

File.AppendAllText(path, c.Name + ": " + ((NumericUpDown)c).Text + "\n");

}

}

}

Таблица 22.4 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| fields values | hr\_info.txt |

Источник: собственная разработка

Анализ результата разработки приложения представлен на рисунке 22.4.

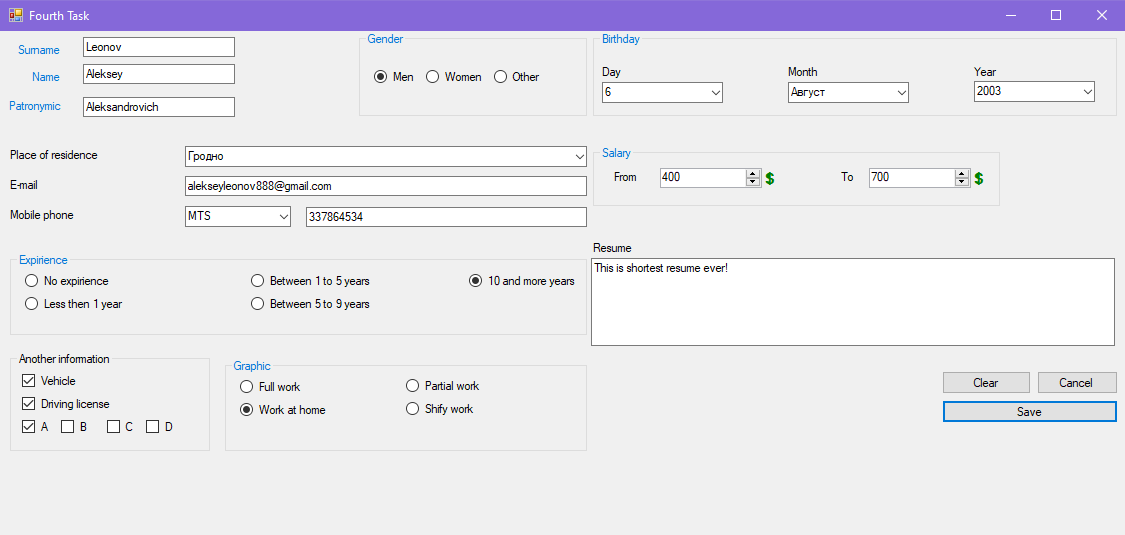


Рисунок 22.4 – Результат работы программы

Источник: собственная разработка