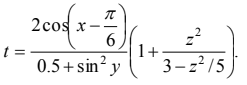
**22 Технология Windows Form. Работа с формами**

Задание №1. Составить программу вычисления для заданных значений x = 14.26, y = -1.22, z = -3.5 \* 10-2 арифметического выражения:



Листинг программы:

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

{

try

{

textBoxX.Text = "14,26";

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

textBoxY.Text = "-1,22";

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

textBoxZ.Text = "0,035";

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

double t = ((2 \* Math.Cos(x - (Math.PI / 6))) / (0.5 \* Math.Pow(Math.Sin(y), 2))) \* (1 + ((Math.Pow(z, 2)) / (3 - (Math.Pow(z, 2) / 5))));

textBoxResult.Text = "День 23;Задание 1;";

textBoxResult.Text += Environment.NewLine + "Result t: " + Convert.ToString(t);

}

catch (Exception ex)

{

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

}

}

Таблица 22.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 14.26, -1.22, 0.035 | -1.7701755 |

Источник: собственная разработка

Анализ результата разработки приложения представлен на рисунке 22.1.

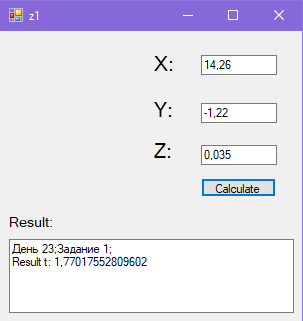
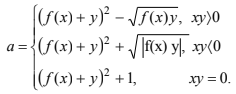


Рисунок 22.1 – Результат работы программы

Источник: собственная разработка

Задание №2. Ввести два числа – x, y. 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 buttonCalculateResult\_Click(object sender, EventArgs e)

{

double result = default(double);

try

{

var x = double.Parse(textBoxX.Text);

var y = double.Parse(textBoxY.Text);

if (x \* y > 0)

{

result = Math.Pow((Math.Pow(x, 2) + y), 2) - Math.Sqrt(Math.Pow(x, 2) \* y);

}

else if (x \* y < 0)

{

result = Math.Pow((Math.Pow(x, 2) + y), 2) + Math.Sqrt(Math.Abs(Math.Pow(x, 2) \* y));

}

else

{

result = Math.Pow((Math.Pow(x, 2) + y), 2) + 1;

}

textBoxResult.Text = "День 23;Задание 2;";

textBoxResult.Text += Environment.NewLine + "Result: " + result;

}

catch (Exception ex)

{

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

}

}

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

{

textBoxX.Text = null;

textBoxY.Text = null;

textBoxResult.Text = null;

}

Таблица 22.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| -4, 4 | 408 |

Источник: собственная разработка

Анализ результата разработки приложения представлен на рисунке 22.2.

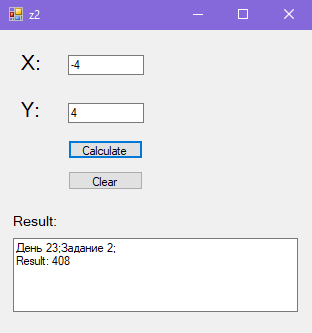


Рисунок 22.2 – Результат работы программы

Источник: собственная разработка

Задание №3. Вычислить и вывести на экран таблицу значений функции y=15.28|x|-3/2 + cos (ln|x| + b) при x, изменяющемся от x0 до xk с шагом dx, a – константа.

Листинг программы:

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

{

double result;

try

{

textBoxResult.ScrollBars = ScrollBars.Vertical;

var x0 = double.Parse(textBoxX0.Text);

var x = x0;

var xK = double.Parse(textBoxXk.Text);

var dX = double.Parse(textBoxDx.Text);

var b = double.Parse(textBoxB.Text);

for (double i = x0; i <= xK; i += dX)

{

result = 15.28 \* Math.Pow(Math.Abs(x), -(3 / 2)) + Math.Cos(Math.Log10(x) + b);

textBoxResult.Text += Convert.ToString(result) + Environment.NewLine;

x += dX;

}

}

catch (Exception ex)

{

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

}

}

Таблица 22.3 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 1, 10, 1, 7 | 16.033, 8.165, 5.461, 2.1917 |

Источник: собственная разработка

Анализ результата разработки приложения представлен на рисунке 22.3.

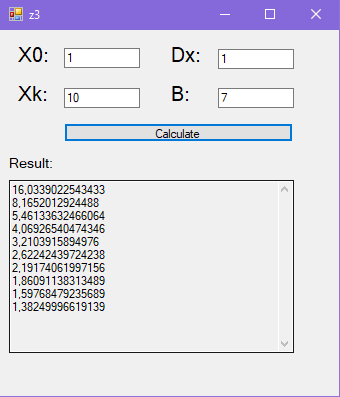


Рисунок 22.3 – Результат работы программы

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)

{

YearBox.Items.Clear();

var thisYear = DateTime.Now.Year;

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

{

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

}

}

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

{

Application.Exit();

}

private static void CleanAllTextBoxesIn(Control parent)

{

foreach (Control c in parent.Controls)

{

if (c is TextBox)

{

((TextBox)c).Clear();

}

if (c.HasChildren)

{

CleanAllTextBoxesIn(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 void button2\_Click(object sender, EventArgs e)

{

CleanAllTextBoxesIn(this);

}

private static void SaveAllTextBoxesIn(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))

{

SaveAllTextBoxesIn(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");

}

}

}

private void button3\_Click(object sender, EventArgs e)

{

File.Delete(@"content.txt");

string path = @"content.txt";

SaveAllTextBoxesIn(this, path);

}

Таблица 22.4 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| значения полей | content.txt |

Источник: собственная разработка

Анализ результата разработки приложения представлен на рисунке 22.4.

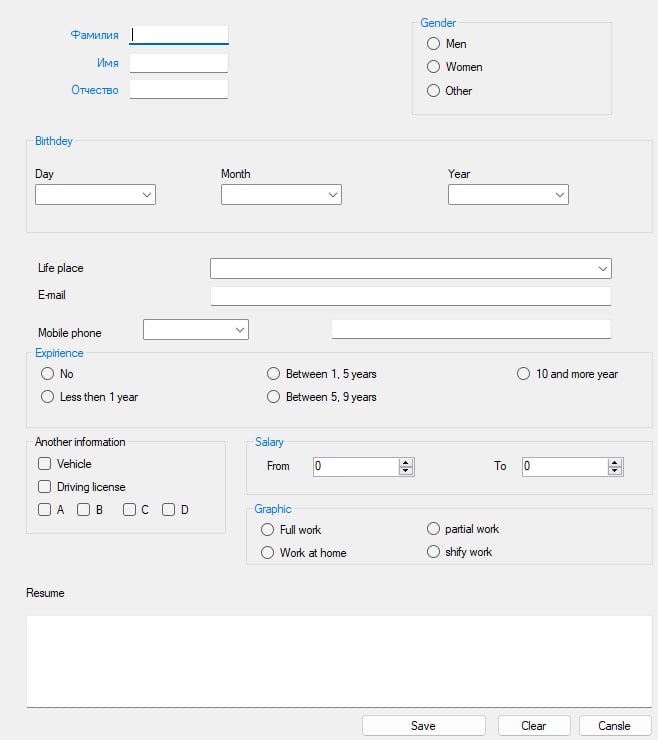


Рисунок 22.4 – Результат работы программы

Источник: собственная разработка