**ПРАКТИЧЕСКОЕ ЗАДАНИЕ 23**

**ЭЛЕМЕНТЫ УПРАВЛЕНИЯ**

Задание 1. Создать программу с кнопкой, меткой и полем ввода. При щелчке на соответствующий элемент на форме динамически должен создаваться подобный ему элемент. Предусмотреть возможность вывода количества кнопок, меток и полей ввода.

Листинг программы:

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

Button button2 = new Button();

button2.Top = button1.Top + button1.Height;

button2.Left = button1.Left;

this.Controls.Add(button2);

}

Анализ результатов:

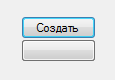


Рисунок 23.1 – Результат работы программы

Источник: собственная разработка

Задание 2. Построить график функции. Таблицу данных получить путём изменения параметра X с шагом h. Самостоятельно выбрать удобные параметры настройки.

Листинг программы:

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

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

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

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

int count = (int)Math.Ceiling((Xmax - Xmin) / Step) + 1;

double[] x = new double[count];

double[] y1 = new double[count];

double[] y2 = new double[count];

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

{

x[i] = Xmin + Step \* i;

y1[i] = Math.Sin(x[i]);

y2[i] = Math.Cos(x[i]);

}

chart1.ChartAreas[0].AxisX.Minimum = Xmin;

chart1.ChartAreas[0].AxisX.Maximum = Xmax;

chart1.ChartAreas[0].AxisX.MajorGrid.Interval = Step;

chart1.Series[0].Points.DataBindXY(x, y1);

chart1.Series[1].Points.DataBindXY(x, y2);

}

}

Таблица 23.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  |  |

Анализ результатов:

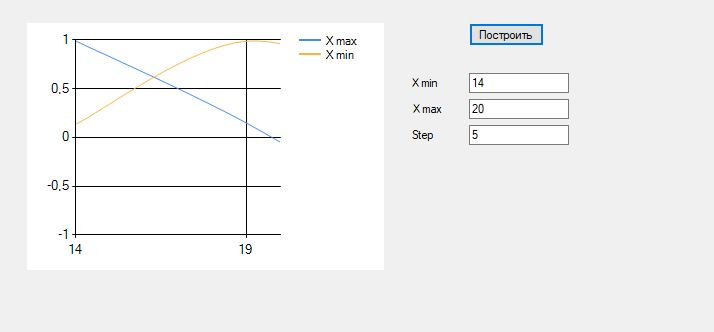


Рисунок 23.2 – Результат работы программы

Источник: собственная разработка

Задание 3. Построить изображение “Компьютер” при помощи фигур.

Листинг программы:

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

Pen pen = new Pen(Color.Black, 2);

private void Form1\_paint(object sender, PaintEventArgs e)

{

Graphics g = e.Graphics;

g.Clear(Color.White);

///клавиатура

g.DrawLine(pen, 250, 250, 250, 260);

g.DrawLine(pen, 250, 260, 350, 260);

g.DrawLine(pen, 350, 260, 350, 250);

g.DrawLine(pen, 350, 250, 250, 250);

///монитор

g.DrawLine(pen, 270, 230, 270, 170);

g.DrawLine(pen, 270, 170, 330, 170);

g.DrawLine(pen, 330, 170, 330, 230);

g.DrawLine(pen, 330, 230, 270, 230);

///ножка

g.DrawLine(pen, 295, 230, 295, 250);

g.DrawLine(pen, 305, 230, 305, 250);

///системник

g.DrawLine(pen, 380, 170, 380, 260);

g.DrawLine(pen, 380, 260, 420, 260);

g.DrawLine(pen, 420, 260, 420, 170);

g.DrawLine(pen, 420, 170, 380, 170);

}

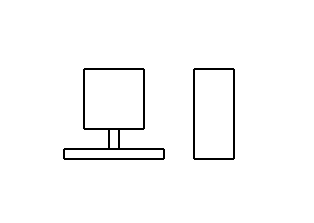


Рисунок 23.3 – Результат работы программы

Источник: собственная разработка