**23 Элементы управления**

Задание 1. Во всех заданиях исходные данные вводить с помощью ListBox. Строки вводятся на этапе проектирования формы, используя окно свойств. Вывод результата организовать в метку Label. Дана строка символов. Вывести на экран количество строчных русских букв, входящих в эту строку.

Листинг программы:

public partial class Form1 : Form{ public Form1(){InitializeComponent();}

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

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

string inputString = string.Empty; int count = 0;

for (int i = 0; i < listBox1.Items.Count; i++) {

inputString += listBox1.Items[i].ToString() + " ";}

foreach (char c in inputString){if (c >= 'а' && c <= 'я'){count++;}}

label1.Text = "Количество строчных русских букв: " + count.ToString();}

Таблица 23.1 – Выходные и входные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| РусскиеИEnglich | Количество строчных русских букв: 6 |

Источник: собственная разработка

Анализ результатов:

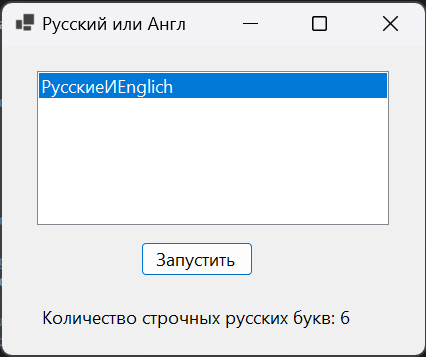


Рисунок 23.1 – Результат работы программы

Источник: собственная разработка

Задание 2. Построить график функции. Таблицу данных получить путём изменения параметра X с шагом h. y = x3.

Листинг программы:

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[] y1 = new double[count];

double[] x = new double[count]; for (int i = 0; i < count; i++){

x[i] = xmin + step \* i; y1[i] = Math.Pow(x[i], 2);}

chart2.ChartAreas[0].AxisX.Minimum = xmin;

chart2.ChartAreas[0].AxisX.Maximum = xmax;

chart2.ChartAreas[0].AxisX.MajorGrid.Interval = step;

chart2.Series[0].Points.DataBindXY(x, y1); }

Таблица 23.2 – Выходные и входные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 1;10;1 |  |

Источник: собственная разработка

Анализ результатов:

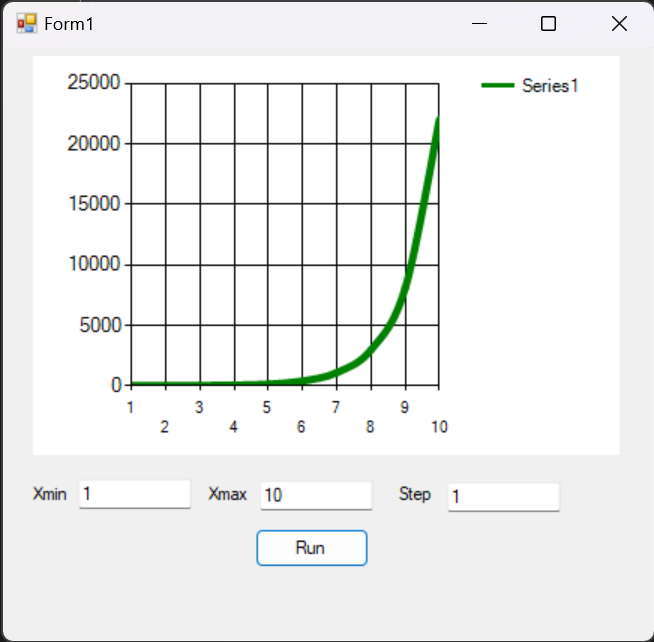


Рисунок 23.2 – Результат работы программы

Источник: собственная разработка

Задание 3. Построить изображение при помощи фигур. Тема: Компьютер.

Листинг программы:

private void Form1\_Paint(object sender, PaintEventArgs e){Graphics g = e.Graphics;

int monitorWidth = 200; int monitorHeight = 160;int monitorX = 50; int monitorY = 50;

g.DrawRectangle(Pens.Black, monitorX, monitorY, monitorWidth, monitorHeight);

g.FillRectangle(Brushes.Black, monitorX + monitorWidth / 2 - 10, monitorY + monitorHeight, 20, 40);

int screenX = monitorX + 10; int screenY = monitorY + 10;

int screenWidth = monitorWidth - 20; int screenHeight = monitorHeight - 20;

g.FillRectangle(Brushes.Black, screenX, screenY, screenWidth, screenHeight);

int logoWidth = 15; int logoHeight = 15;

int logoX = screenX + screenWidth / 2 - logoWidth / 2;

int logoY = screenY + screenHeight / 2 - logoHeight / 2; int logoPadding = 2;

g.FillRectangle(Brushes.White, logoX, logoY, logoWidth, logoHeight);

g.FillRectangle(Brushes.White, logoX + logoWidth + logoPadding, logoY, logoWidth, logoHeight);

g.FillRectangle(Brushes.White, logoX, logoY + logoHeight + logoPadding, logoWidth, logoHeight);

g.FillRectangle(Brushes.White, logoX + logoWidth + logoPadding, logoY + logoHeight + logoPadding, logoWidth, logoHeight);}

Таблица 23.3 – Выходные и входные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | Изображение компьютера |

Источник: собственная разработка

Анализ результатов:

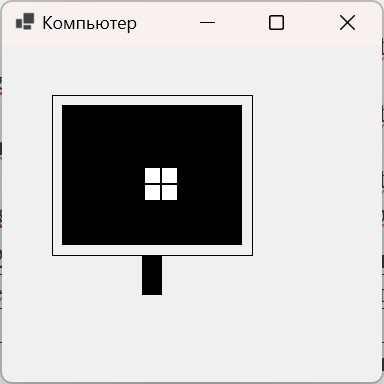


Рисунок 23.3 – Результат работы программы

Источник: собственная разработка

Задание 4. Написать программу с кнопкой и тремя полями ввода. При нажатии на кнопку программа анализирует содержимое первого поля и динамически порождает элемент управления. Если в первом поле ввода содержится буква «К», то на форму добавляется кнопка, если буква «П»-поле ввода, если «М»-метка. Во втором и третьем поле находятся координаты левого верхнего угла будущего элемента управления. При создании будущего элемента настроить его свойства (на ваш выбор (цвет, размер и т.п)).

При наведении курсора на элемент управления он должен быть удален с формы.

Листинг программы:

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

{string litter = textBox1.Text;

if (litter == "К" || litter == "к"){button1.Visible = true;}

else if (litter == "П" || litter == "п") {textBox4.Visible = true;}

else if (litter == "М" || litter == "м") {label3.Visible = true;}

else{MessageBox.Show("Не поддерживаемая буква");}}

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

{textBox1.Clear();}

Таблица 23.4 – Выходные и входные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| м, к, п | Появление элементов на поверхности окна. |

Источник: собственная разработка

Анализ результатов:

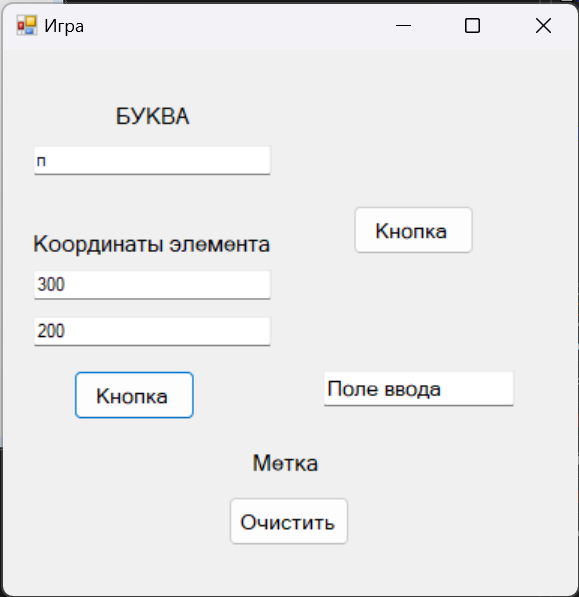


Рисунок 23.4 – Результат работы программы

Источник: собственная разработка