МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Казанский национальный исследовательский технический университет

им. А.Н. Туполева – КАИ»

Институт компьютерных технологий и защиты информации

Отделение СПО ИКТЗИ (Колледж информационных технологий)

ЛАБОРАТОРНАЯ РАБОТА №7

по дисциплине

Основы алгоритмизации и программирования

Тема: «Введение в разработку графических пользовательских интерфейсов с использованием технологии Windows Forms»

Работу выполнил

Студент гр.4238

Бусов В.Р.

Принял

Преподаватель Шмидт. И.Р.

Казань 2023

**Варианты 4 4 4 4 4**

**Цель работы**

Научиться размещать и настраивать внешний вид элементов управления на форме и создавать обработчики событий.

**Задание на лабораторную работу**

Выбрать пять заданий из лабораторных работ №2-6 и решить их, разработав графический интерфейс с использованием технологии Windows Forms.

Лр2\_зд1: разработать тест из 5 и боле вопросов.

Лр3\_зд1: дано натуральное число N. Найти все натуральные числа меньшие N, которые являются палиндромами

Лр4\_зд1: Дана действительная матрица размерности 𝑚 × 𝑛, в которой ее все элементы равны нулю. Получить новую матрицу путем деления всех элементов данной матрицы на е наибольший по модулю элемент

Лр5\_зд2: написать программу, выполняющую расчет суммы двух неквадратных матриц. Матрицы должны храниться в памяти в виде двумерного динамического массива, размерности матриц вводятся пользователем с клавиатуры. Предусмотреть генерацию значений матриц как случайных вещественных чисел в диапазоне от -1.0 до 1.0.

Лр6\_зд1: реализовать класс Account, представляющий собой банковский счет. В классе должны быть реализованы 4 поля: фамилия владельца, номер счета, процент начисления и сумма в рублях. Необходимо выполнять следующие операции: сменить владельца счета, снять некоторую сумму со счета, положить деньги на счет, начислить проценты, перевести сумму в доллары, перевести сумму в евро.

**Результат выполнения работы**

Для выполнения всех заданий мне понадобились такие элементы, как button - кнопка, textbox - поле для ввода значений, listbox - поле для вывода массивов, label использовался для вывода результатов и инструкций к вводу данных. Главная форма, с помощью которой осуществляется переход к другим показана на рисунке 1. Формы 2-6 лабораторных показаны на рисунках 2-6 соответственно.

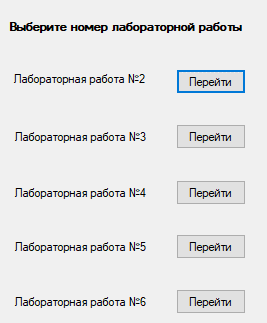


Рисунок 1 - Главная форма

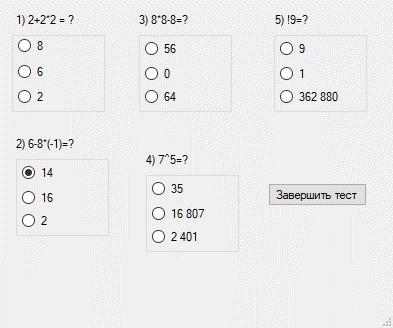


Рисунок 2 - Форма второй лабораторной работы

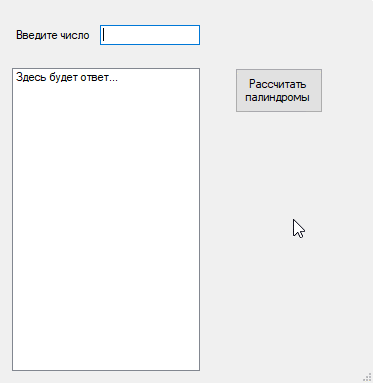


Рисунок 3 - Форма третьей лабораторной работы

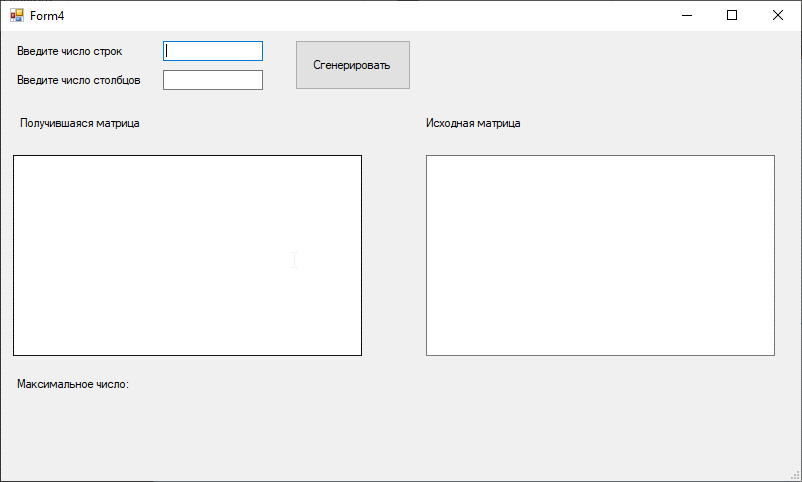


Рисунок 4 - Форма четвёртой лабораторной работы

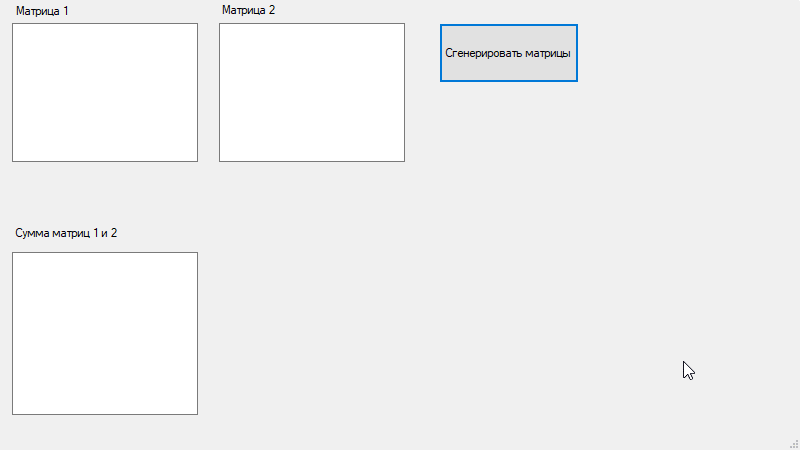


Рисунок 5 - Форма пятой лабораторной работы

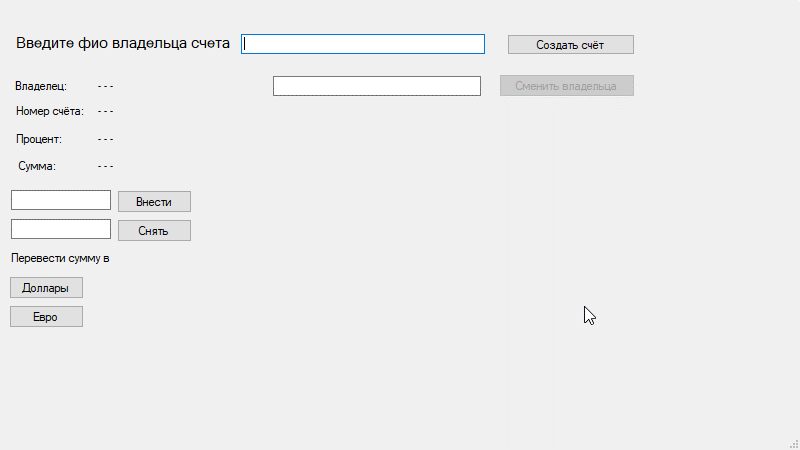


Рисунок 6 - Форма шестой лабораторной работы

**Задание 1**

На рисунке 7 можно увидеть приветственное сообщение, которое объясняет смысл задания. После нажатия на кнопку «ОК» мы переходим к окну взаимодействия. Далее необходимо выбрать все варианты ответов в тесте. После выбора и нажатия на кнопку «Завершить тест» мы увидим результат. Правильные ответы подсвечиваются зеленым цветом, а неправильные – красным (Рисунок 8)

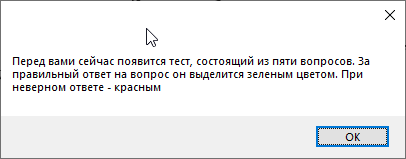


Рисунок 7 – Приветственное сообщение



Рисунок 8 – Результат теста

**Задание 2**

На рисунке 9 можно увидеть приветственное сообщение, которое объясняет смысл задания. После нажатия на кнопку «ОК» мы переходим к окну взаимодействия. В окне взаимодействия мы должны ввести число, нажать на кнопку «Рассчитать палиндромы», после чего в ListBox-е программа нам выведет все числа-палиндромы, меньшие введенного числа. Пример представлен ниже на рисунке 10.

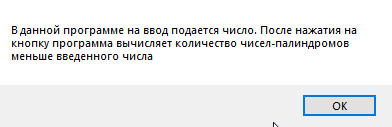


Рисунок 9 – Приветственное сообщение

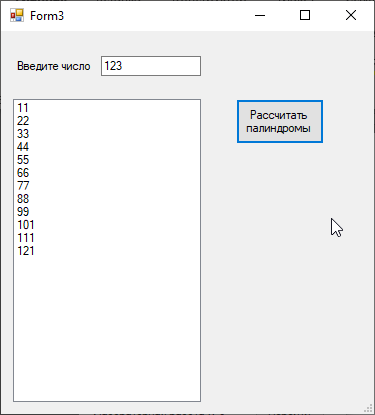


Рисунок 10 – Результат выполнения программы по поиску чисел-палиндромов

**Задание 3**

На рисунке 11 можно увидеть приветственное сообщение, которое объясняет смысл задания. После нажатия на кнопку «ОК» мы переходим к окну взаимодействия. В окне необходимо ввести два числа: количество столбцов и количество строк в матрице. Далее нажимается кнопка «Сгенерировать», после чего происходит генерация исходной матрицы и вычисляется результирующая матрица (Рисунок 13)

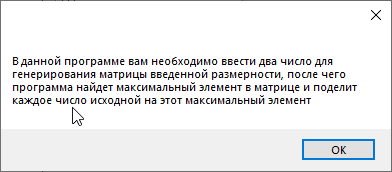


Рисунок 11 – Приветственное сообщение

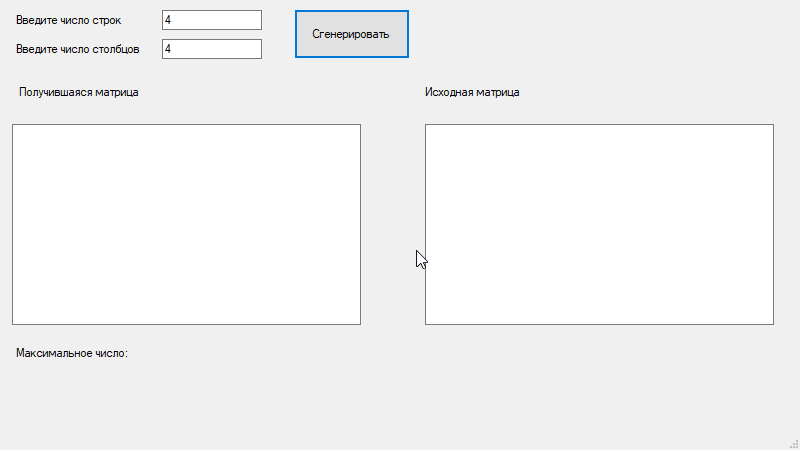


Рисунок 12 – Ввод количества строк и количества столбцов

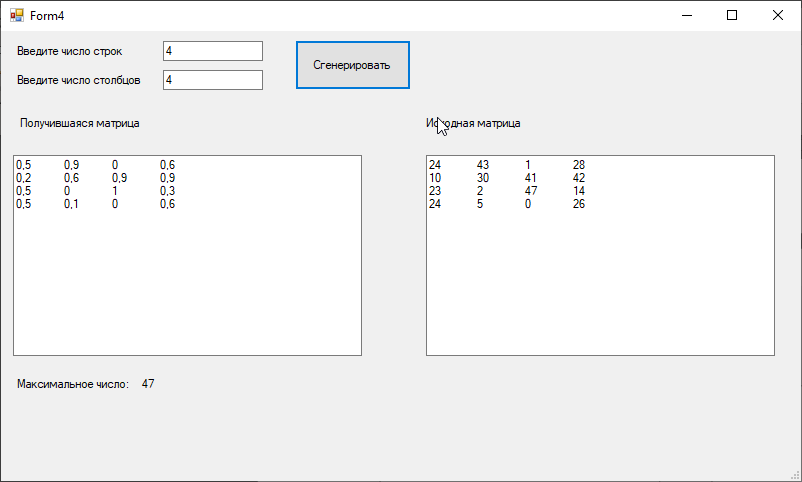


Рисунок 13 – Результат работы программы. В данном случае максимальное число в исходной матрице было 47. Получавшаяся матрица была сгенерирована путем деления исходной на максимальное число

**Задание 4**

На рисунке 14 можно увидеть приветственное сообщение, которое объясняет смысл задания. После нажатия на кнопку «ОК» мы переходим к окну взаимодействия. В окне взаимодействия необходимо нажать кнопку «Сгенерировать матрицы» для создания двух матриц одинаковой разрядности.

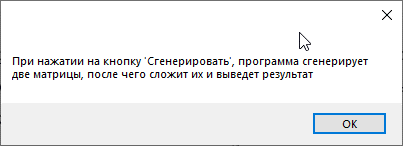


Рисунок 14 – Приветственное сообщение

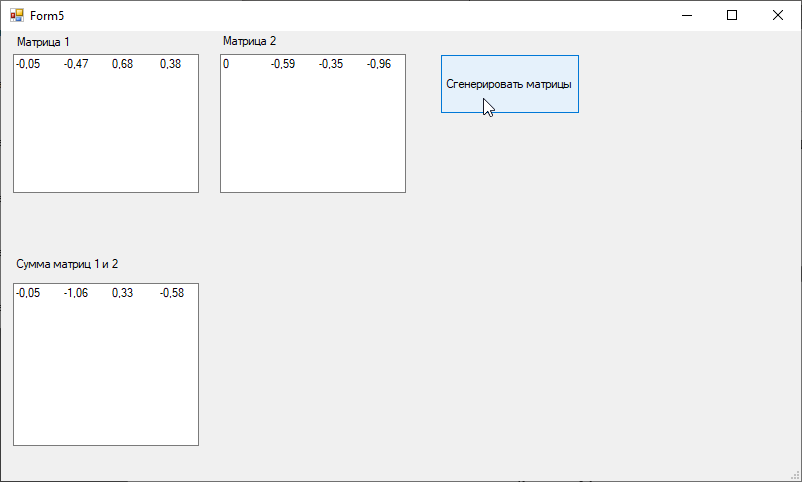


Рисунок 15 – Результат работы программы

**Задание 5**

На рисунке 16 можно увидеть приветственное сообщение, которое объясняет смысл задания. После нажатия на кнопку «ОК» мы переходим к окну взаимодействия. В окне взаимодействия сначала необходимо ввести ФИО владельца счет (Рисунок 17), после чего нажать на кнопку «Создать счет». Далее нам откроются все методы взаимодействия со счетом: зачисления (Рисунок 18), снятие (Рисунок 19), перевод суммы в доллары и евро (Рисунок 20) и смена владельца счета (Рисунок 21).

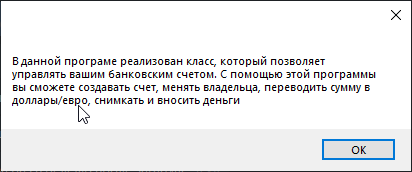


Рисунок 16 – Приветственное сообщение

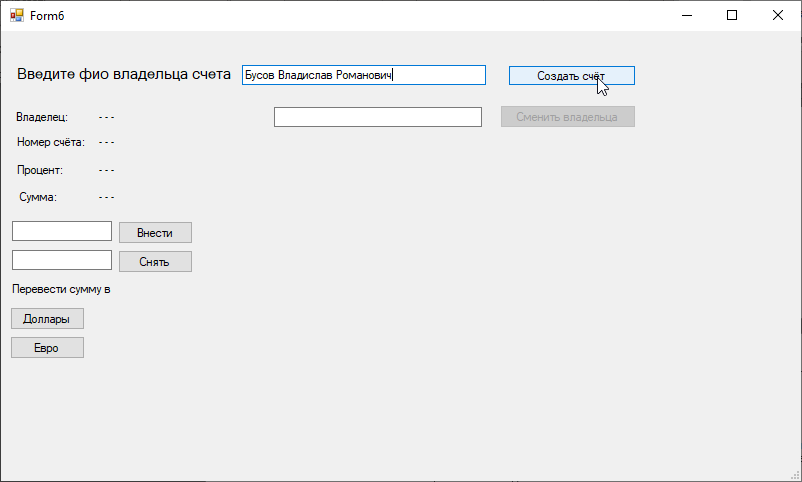


Рисунок 17 – Создание счета

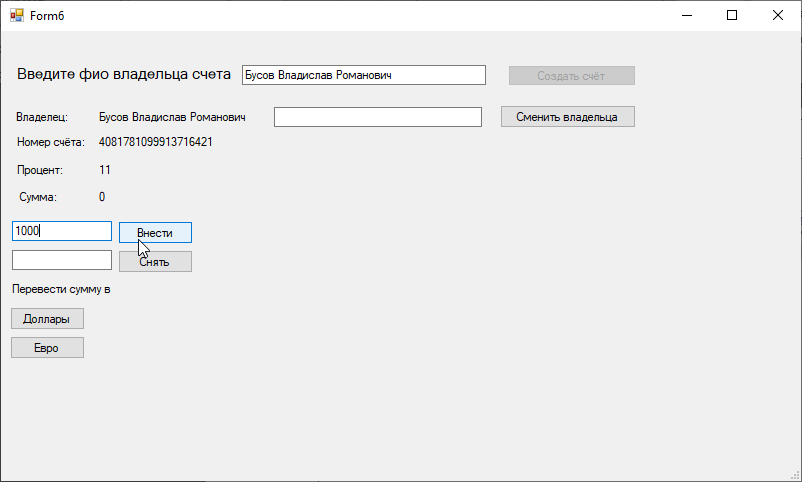


Рисунок 18 – Внесение средств на счет

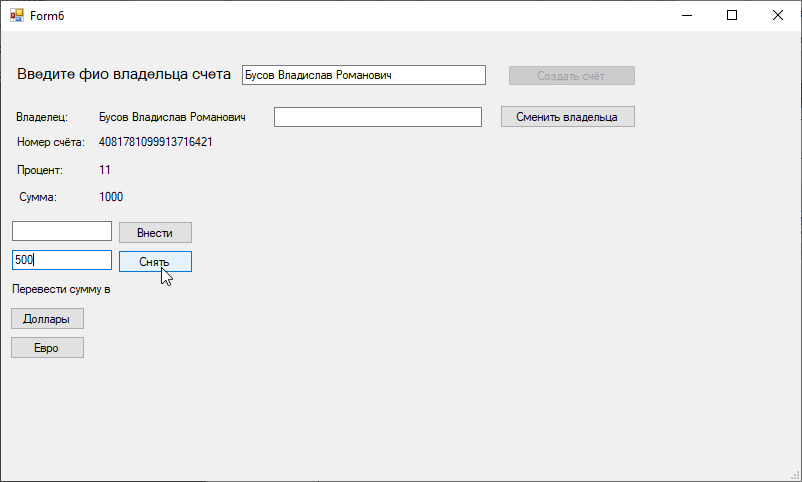


Рисунок 19 – Снятие средств со счета

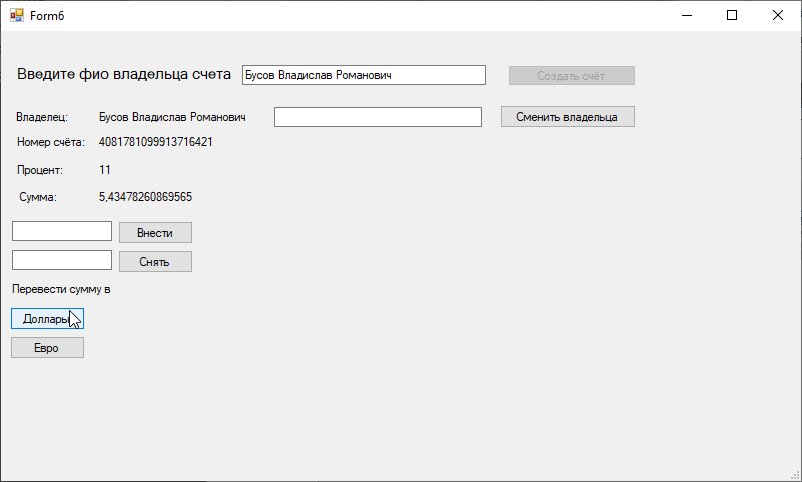


Рисунок 20 – Перевод суммы из рублей в доллары

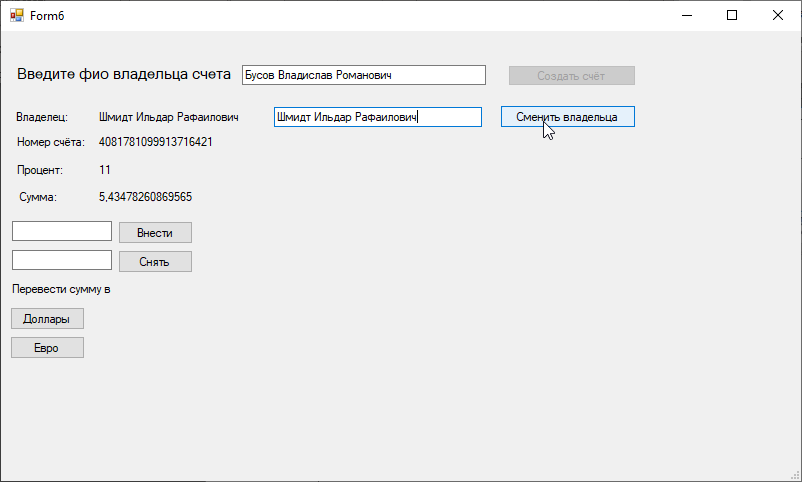


Рисунок 21 – Смена владельца счета

**Листинг программ**

**Form1.cs**

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 WindowsFormsApp1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

Form2 frm = new Form2();

MessageBox.Show("Перед вами сейчас появится тест, состоящий из пяти вопросов. За правильный ответ на вопрос он выделится зеленым цветом. При неверном ответе - красным");

frm.ShowDialog();

}

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

{

Form3 frm = new Form3();

MessageBox.Show("В данной программе на ввод подается число. После нажатия на кнопку программа вычисляет количество чисел-палиндромов меньше введенного числа");

frm.ShowDialog();

}

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

{

Form4 frm = new Form4();

MessageBox.Show("В данной программе вам необходимо ввести два число для генерирования матрицы введенной размерности, после чего программа найдет максимальный элемент в матрице и поделит каждое число исходной на этот максимальный элемент");

frm.ShowDialog();

}

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

{

Form5 frm = new Form5();

MessageBox.Show("При нажатии на кнопку 'Сгенерировать', программа сгенерирует две матрицы, после чего сложит их и выведет результат");

frm.ShowDialog();

}

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

{

Form6 frm = new Form6();

MessageBox.Show("В данной програме реализован класс, который позволяет управлять вашим банковским счетом. С помощью этой программы вы сможете создавать счет, менять владельца, переводить сумму в доллары/евро, снимкать и вносить деньги");

frm.ShowDialog();

}

}

}

**Form2.cs**

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 WindowsFormsApp1

{

public partial class Form2 : Form

{

public Form2()

{

InitializeComponent();

}

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

{

if (radioButton2.Checked) groupBox1.BackColor = Color.Green;

else groupBox1.BackColor = Color.Red;

if (radioButton5.Checked) groupBox2.BackColor = Color.Green;

else groupBox2.BackColor = Color.Red;

if (radioButton8.Checked) groupBox3.BackColor = Color.Green;

else groupBox3.BackColor = Color.Red;

if (radioButton10.Checked) groupBox4.BackColor = Color.Green;

else groupBox4.BackColor = Color.Red;

if (radioButton15.Checked) groupBox5.BackColor = Color.Green;

else groupBox5.BackColor = Color.Red;

}

}

}

**Form3.cs**

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 WindowsFormsApp1

{

public partial class Form3 : Form

{

public Form3()

{

InitializeComponent();

}

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

{

if (int.TryParse(textBox1.Text, out int N))

{

listBox1.Items.Clear();

if (N < 11) listBox1.Items.Add("Палиндромов, меньше числа N не найдено");

else

{

for (int i = 11; i <= N; i++)

{

string num = Convert.ToString(i);

var reversed\_num = new string(num.Reverse().ToArray());

if (reversed\_num.Equals(num))

{

listBox1.Items.Add(num);

}

}

}

} else MessageBox.Show("Ошибка вводных данных! На вход я могу принимать только целые числа");

}

}

}

**Form4.cs**

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 WindowsFormsApp1

{

public partial class Form4 : Form

{

public Form4()

{

InitializeComponent();

}

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

{

if (int.TryParse(textBox1.Text, out int rows))

{

if (int.TryParse(textBox2.Text, out int cols))

{

if (!(rows <= 6 && cols <= 6)) {

MessageBox.Show("Пожалуйста, введите числа не более 6");

} else {

textBox3.Text = "";

textBox4.Text = "";

var rand = new Random();

double[,] matrix = new double[rows, cols];

double max\_n = 0;

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

{

for (int j = 0; j < cols; j++)

{

matrix[i, j] = rand.Next(50);

textBox4.Text += $"{matrix[i, j]}\t";

if (matrix[i, j] > max\_n) max\_n = matrix[i, j];

}

textBox4.Text += "\r\n";

}

label4.Text = max\_n.ToString();

for (int row = 0; row < matrix.GetUpperBound(0) + 1; row++)

{

for (int col = 0; col < matrix.GetUpperBound(1) + 1; col++)

{

matrix[row, col] = Math.Round(matrix[row, col] / max\_n, 1);

textBox3.Text += $"{matrix[row, col]}\t";

}

textBox3.Text += "\r\n";

}

}

} else {

MessageBox.Show("Введите количество столбцов в формате целого числа");

}

} else {

MessageBox.Show("Введите количество строк в формате целого числа");

}

}

}

}

**Form5.cs**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Reflection.Emit;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using static System.Windows.Forms.VisualStyles.VisualStyleElement;

namespace WindowsFormsApp1

{

public partial class Form5 : Form

{

public Form5()

{

InitializeComponent();

}

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

{

textBox1.Text = "";

textBox2.Text = "";

textBox3.Text = "";

var rand = new Random();

int cols = rand.Next(1, 5);

int rows = rand.Next(1, 5);

double[,] matrix1 = new double[rows, cols];

double[,] matrix2 = new double[rows, cols];

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

{

for (int j = 0; j < cols; j++)

{

matrix1[i, j] = Math.Round(rand.NextDouble() \* 2 - 1, 2);

matrix2[i, j] = Math.Round(rand.NextDouble() \* 2 - 1, 2);

textBox1.Text += $"{matrix1[i, j]}\t";

textBox2.Text += $"{matrix2[i, j]}\t";

}

textBox1.Text += "\r\n";

textBox2.Text += "\r\n";

}

for (int row = 0; row < matrix1.GetUpperBound(0) + 1; row++)

{

for (int col = 0; col < matrix1.GetUpperBound(1) + 1; col++)

{

textBox3.Text += $"{matrix1[row, col] + matrix2[row, col]}\t";

}

textBox3.Text += "\r\n";

}

}

}

}

**Form6.cs**

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 WindowsFormsApp1

{

public partial class Form6 : Form

{

Accaunt acc;

public Form6()

{

InitializeComponent();

button6.Enabled = false;

}

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

{

button1.Enabled = false;

button6.Enabled = true;

this.acc = new Accaunt(textBox1.Text);

label10.Text = acc.get\_owner();

label3.Text = acc.get\_acc\_num().ToString();

label5.Text = acc.get\_percent().ToString();

label7.Text = acc.get\_summ().ToString();

}

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

{

if (int.TryParse(textBox2.Text, out int summ))

{

bool res = this.acc.put\_summ(summ);

if (!res) MessageBox.Show("Ошибка! Введено отрицательное число");

else label7.Text = this.acc.get\_summ().ToString();

} else {

MessageBox.Show("Ошибка! Неверный формат входных данных");

}

textBox2.Text = "";

}

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

{

if (int.TryParse(textBox3.Text, out int summ))

{

bool res = this.acc.take\_summ(summ);

if (!res) MessageBox.Show("Ошибка! Проверьте, что вы ввели");

else label7.Text = this.acc.get\_summ().ToString();

}

else

{

MessageBox.Show("Ошибка! Неверный формат входных данных");

}

textBox3.Text = "";

}

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

{

label7.Text = this.acc.to\_dollars().ToString();

}

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

{

label7.Text = this.acc.to\_euros().ToString();

}

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

{

this.acc.set\_owner(textBox4.Text);

textBox4.Text = "";

label10.Text = this.acc.get\_owner();

}

}

class Accaunt

{

private string owner;

private double summ;

private long acc\_num;

private int percent;

public Accaunt(string owner)

{

this.owner = owner;

var rand = new Random();

this.summ = 0;

this.acc\_num = Convert.ToInt64($"408178109991{rand.Next(1000000, 9999999)}");

this.percent = rand.Next(5, 15);

}

public long get\_acc\_num()

{

return this.acc\_num;

}

public int get\_percent()

{

return this.percent;

}

public void set\_owner(string owner)

{

this.owner = owner;

}

public string get\_owner()

{

return this.owner;

}

public bool take\_summ(double delta)

{

if (delta <= this.summ && delta > 0)

{

this.summ -= delta;

return true;

}

return false;

}

public bool put\_summ(double delta)

{

if (delta >= 0)

{

this.summ += delta;

return true;

}

else

{

return false;

}

}

public void add\_percent()

{

this.summ += (this.summ / 100) \* (this.percent / 12);

}

public double get\_summ()

{

return this.summ;

}

public double to\_dollars()

{

return this.summ / 92;

}

public double to\_euros()

{

return this.summ / 99;

}

}

}