**Лабораторная работа №4**

**Вариант №14**

**Тема «Массивы. Алгоритмы обработки массивов. Рекурсия»**

Дан массив A размера N. Вывести его элементы в следующем порядке:

A1, AN, A2, AN–1, A3, AN–2, ... .

**Код программы:**

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 PR4

{

public partial class Form1 : Form

{

private int[,] matrix = new int[8, 5]; // матрица 8x5

public Form1()

{

InitializeComponent();

}

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

{

// Парсим введенную пользователем строку с числами в массив int

int[] A = Array.ConvertAll(textBox1.Text.Split(' '), int.Parse);

int n = A.Length;

listBox1.Items.Clear();

for (int i = 0; i < n / 2; i++)

{

listBox1.Items.Add(A[i]);

listBox1.Items.Add(A[n - i - 1]);

}

if (n % 2 != 0)

{

listBox1.Items.Add(A[n / 2]);

}

}

private void tabPage2\_Click(object sender, EventArgs e)

{

}

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

{

// Парсим введенную пользователем строку с числами в массив int

int[] arr = Array.ConvertAll(textBox2.Text.Split(' '), int.Parse);

// Ищем первый локальный минимум

int n = arr.Length;

for (int i = 1; i < n - 1; i++)

{

if (arr[i] < arr[i - 1] && arr[i] < arr[i + 1])

{

// Нашли локальный минимум, выводим его индекс и выходим из цикла

label3.Text = "Первый локальный минимум: " + i.ToString();

return;

}

}

// Если локальный минимум не найден, выводим соответствующее сообщение

label3.Text = "Локальный минимум не найден";

}

private void label4\_Click(object sender, EventArgs e)

{

}

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

{

Random rand = new Random();

// заполнение матрицы случайными числами

for (int i = 0; i < 8; i++)

{

for (int j = 0; j < 5; j++)

{

matrix[i, j] = rand.Next(20, 41);

}

}

// вывод матрицы в textBox

textBox3.Clear();

for (int i = 0; i < 8; i++)

{

for (int j = 0; j < 5; j++)

{

textBox3.Text += matrix[i, j].ToString() + " ";

}

textBox3.Text += Environment.NewLine;

}

// поиск строки с не более чем 3 четными числами

int row = -1;

for (int i = 0; i < 8; i++)

{

int count = 0;

for (int j = 0; j < 5; j++)

{

if (matrix[i, j] % 2 == 0)

{

count++;

}

}

if (count <= 3)

{

row = i;

break;

}

}

// вывод результата

if (row == -1)

{

textBox3.Text += "Строки с не более чем 3 четными числами не найдено";

}

else

{

textBox3.Text += "Номер строки с не более чем 3 четными числами: " + (row + 1).ToString();

}

}

private void tabPage1\_Click(object sender, EventArgs e)

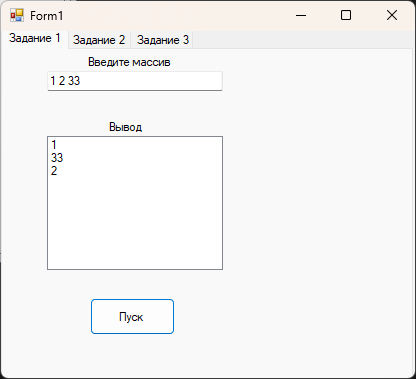
{

}

}

}

**Тестирование:**



Индивидуальные задания №2

## Анализ элементов массива и его вывод

В заданиях предполагается, что размер массива не превосходит 10. Индекс начального элемента массива считается равным 0.

**Вариант 15.**

Дан массив размера N. Найти номер его первого локального минимума

(локальный минимум — это элемент, который меньше любого из своих

соседей).

**Код программы:**

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 PR4

{

public partial class Form1 : Form

{

private int[,] matrix = new int[8, 5]; // матрица 8x5

public Form1()

{

InitializeComponent();

}

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

{

// Парсим введенную пользователем строку с числами в массив int

int[] A = Array.ConvertAll(textBox1.Text.Split(' '), int.Parse);

int n = A.Length;

listBox1.Items.Clear();

for (int i = 0; i < n / 2; i++)

{

listBox1.Items.Add(A[i]);

listBox1.Items.Add(A[n - i - 1]);

}

if (n % 2 != 0)

{

listBox1.Items.Add(A[n / 2]);

}

}

private void tabPage2\_Click(object sender, EventArgs e)

{

}

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

{

// Парсим введенную пользователем строку с числами в массив int

int[] arr = Array.ConvertAll(textBox2.Text.Split(' '), int.Parse);

// Ищем первый локальный минимум

int n = arr.Length;

for (int i = 1; i < n - 1; i++)

{

if (arr[i] < arr[i - 1] && arr[i] < arr[i + 1])

{

// Нашли локальный минимум, выводим его индекс и выходим из цикла

label3.Text = "Первый локальный минимум: " + i.ToString();

return;

}

}

// Если локальный минимум не найден, выводим соответствующее сообщение

label3.Text = "Локальный минимум не найден";

}

private void label4\_Click(object sender, EventArgs e)

{

}

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

{

Random rand = new Random();

// заполнение матрицы случайными числами

for (int i = 0; i < 8; i++)

{

for (int j = 0; j < 5; j++)

{

matrix[i, j] = rand.Next(20, 41);

}

}

// вывод матрицы в textBox

textBox3.Clear();

for (int i = 0; i < 8; i++)

{

for (int j = 0; j < 5; j++)

{

textBox3.Text += matrix[i, j].ToString() + " ";

}

textBox3.Text += Environment.NewLine;

}

// поиск строки с не более чем 3 четными числами

int row = -1;

for (int i = 0; i < 8; i++)

{

int count = 0;

for (int j = 0; j < 5; j++)

{

if (matrix[i, j] % 2 == 0)

{

count++;

}

}

if (count <= 3)

{

row = i;

break;

}

}

// вывод результата

if (row == -1)

{

textBox3.Text += "Строки с не более чем 3 четными числами не найдено";

}

else

{

textBox3.Text += "Номер строки с не более чем 3 четными числами: " + (row + 1).ToString();

}

}

private void tabPage1\_Click(object sender, EventArgs e)

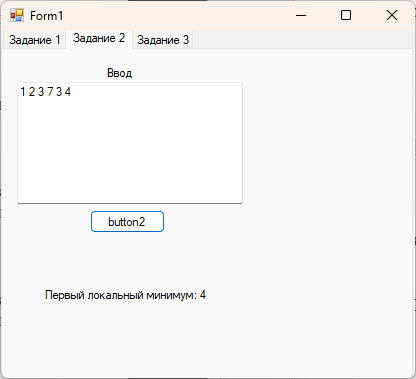
{

}

}

}

**Тестирование:**



Индивидуальные задания №3

Вывести на экран матрицу 8x5, элементами которой являются целые

случайные числа из интервала [20,40]. Определить номер строки,

содержащей не более 3-х чётных чисел.

**Код программы:**

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 PR4

{

public partial class Form1 : Form

{

private int[,] matrix = new int[8, 5]; // матрица 8x5

public Form1()

{

InitializeComponent();

}

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

{

// Парсим введенную пользователем строку с числами в массив int

int[] A = Array.ConvertAll(textBox1.Text.Split(' '), int.Parse);

int n = A.Length;

listBox1.Items.Clear();

for (int i = 0; i < n / 2; i++)

{

listBox1.Items.Add(A[i]);

listBox1.Items.Add(A[n - i - 1]);

}

if (n % 2 != 0)

{

listBox1.Items.Add(A[n / 2]);

}

}

private void tabPage2\_Click(object sender, EventArgs e)

{

}

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

{

// Парсим введенную пользователем строку с числами в массив int

int[] arr = Array.ConvertAll(textBox2.Text.Split(' '), int.Parse);

// Ищем первый локальный минимум

int n = arr.Length;

for (int i = 1; i < n - 1; i++)

{

if (arr[i] < arr[i - 1] && arr[i] < arr[i + 1])

{

// Нашли локальный минимум, выводим его индекс и выходим из цикла

label3.Text = "Первый локальный минимум: " + i.ToString();

return;

}

}

// Если локальный минимум не найден, выводим соответствующее сообщение

label3.Text = "Локальный минимум не найден";

}

private void label4\_Click(object sender, EventArgs e)

{

}

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

{

Random rand = new Random();

// заполнение матрицы случайными числами

for (int i = 0; i < 8; i++)

{

for (int j = 0; j < 5; j++)

{

matrix[i, j] = rand.Next(20, 41);

}

}

// вывод матрицы в textBox

textBox3.Clear();

for (int i = 0; i < 8; i++)

{

for (int j = 0; j < 5; j++)

{

textBox3.Text += matrix[i, j].ToString() + " ";

}

textBox3.Text += Environment.NewLine;

}

// поиск строки с не более чем 3 четными числами

int row = -1;

for (int i = 0; i < 8; i++)

{

int count = 0;

for (int j = 0; j < 5; j++)

{

if (matrix[i, j] % 2 == 0)

{

count++;

}

}

if (count <= 3)

{

row = i;

break;

}

}

// вывод результата

if (row == -1)

{

textBox3.Text += "Строки с не более чем 3 четными числами не найдено";

}

else

{

textBox3.Text += "Номер строки с не более чем 3 четными числами: " + (row + 1).ToString();

}

}

private void tabPage1\_Click(object sender, EventArgs e)

{

}

}

}

**Тестирование:**

