Лабораторная работа №3

«Работа с одномерными массивами»

**Задание**

Дан динамический целочисленный массив. Найти среднее геометрическое всех четных элементов. Разработайте функцию, которая формирует новый массив, их тех элементов исходного массива, значения которых больше найденного количества четных элементов.

Требуется создать приложение, запрашивающее у пользователя через **InputBox** длину массива, а также минимально и максимально возможные значения, которыми он будет заполнен. После надо создать новый массив, содержащий в себе числа, удовлетворяющие заданию.

Также необходимо разработать кнопки, которые будут: выводить результат в текстовый файл, выводить результат в Excel файл и выводить результат в Word файл.

Для выполнения данного задания были разработаны следующие методы:

public static void Enter\_massiv(int[] mas, int length, int min, int max) – Метод для заполнения массива случайными числами с ограничениями которые передаются в качестве аргументов, первым передается сам массив который мы будем заполнять, вторым параметром длина этого массива, после передается минимальный и максимальный порог генерации случайных чисел.

public static void Output\_massiv(int[] mas, int length, DataGridView Table) – Метод для вывода массива в таблицу DataGridView, принимает сам массив, его длину и таблицу для вывода.

public static void Sem2\_Lab3\_Calc\_Geometric(int[] mas, out double result) – Метод для вычисления среднего геометрического. Принимает массив и выводит через out результат.

public static void Sem2\_Lab3\_Count\_Condition(int[] mas, double result, out int index) – Метод для подсчета количества чисел которые подходят под условие задачи. Принимает массив, средне геометрическое вещественного типа и выводит результат через out.

public static void Sem2\_Lab3\_Calculate(int[] mas, double result, int index, out int[] output) – Метод для заполнения нового массива числами подходящими под условиями. Принимает массива с исходными числами, средне геометрическое вещественного типа, количество чисел подходящих под условие, требуется для создания нового массива определенной размерности, вывод

public static void Sem2\_Lab3\_Save\_Txt(int[] array, int[] array2) – Метод для вывода двух передаваемых массивов в текстовый файл.

public static void Sem2\_Lab3\_Save\_Excel(int[] array, int[] array2) – Метод для вывода двух передаваемых массивов в excel файл.

public static void Sem2\_Lab3\_Save\_Word(int[] array, int[] array2) – Метод для вывода двух передаваемых массивов в word файл.

Все методы находятся в DLL-библиотеке.

* 1. **Разработка алгоритма решения задачи.**

Схема алгоритма событийной процедуры решения задачи представлена на рисунке 1.

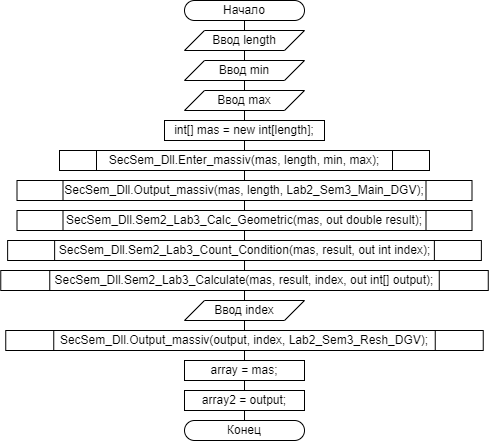


Рисунок 1 – Схема алгоритма процедуры

Схема заполнения массива с заданными пользователем условиями представлена на рисунке 2.

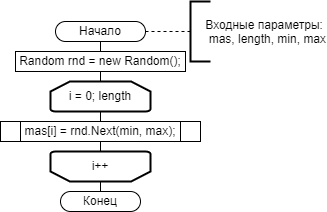


Рисунок 2 – Заполнение массива.

Схема алгоритма вывода массива в таблицу DataGridView представлена на рисунке 3.

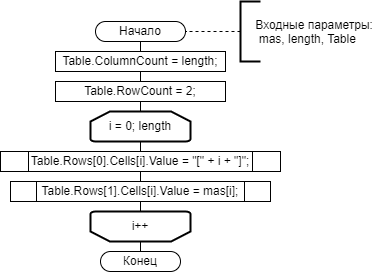


Рисунок 3 – Вывод массива в таблицу.

Схема алгоритма расчёта среднего геометрического всех четных чисел представлена на рисунке 4.

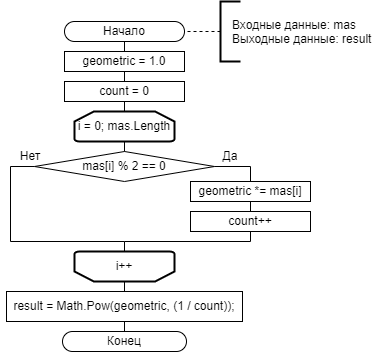


Рисунок 4 – Схема расчёта среднего геометрического.

Схема алгоритма который считает количество чисел которые подходят под условие представлена на рисунке 5.

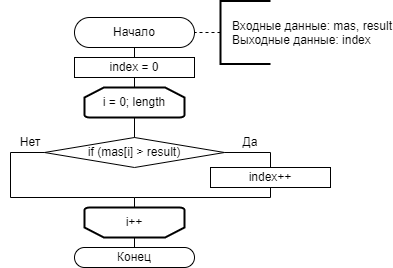


Рисунок 5 – Схема счёта чисел.

Схема алгоритма заполнения нового массива числами которые подходят под условие представлена на рисунке 6.

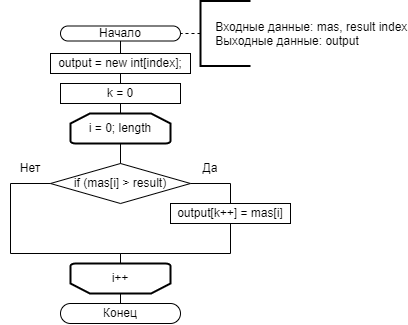


Рисунок 6 – Схема заполнения массива.

* 1. **Разработка программного кода.**

Методы, находящиеся в DLL-библиотеке:

using System;

using System.Windows.Forms;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Diagnostics;

using System.IO;

using Excel = Microsoft.Office.Interop.Excel;

using Word = Microsoft.Office.Interop.Word;

using Microsoft.Office.Interop.Excel;

using System.Security.Cryptography.X509Certificates;

using Microsoft.Office.Interop.Word;

namespace Second\_Semestr\_Dll

{

public class SecSem\_Dll

{

public static void Enter\_massiv(int[] mas, int length, int min, int max)

{

// Убрать length так как легко крашнуть

Random rnd = new Random();

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

mas[i] = rnd.Next(min, max);

}

public static void Output\_massiv(int[] mas, int length, DataGridView Table)

{

Table.ColumnCount = length;

Table.RowCount = 2;

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

{

Table.Rows[0].Cells[i].Value = "[" + i + "]";

Table.Rows[1].Cells[i].Value = mas[i];

}

}

public static void Sem2\_Lab3\_Calc\_Geometric(int[] mas, out double result)

{

double geometric = 1.0;

double count = 0;

for (int i = 0; i < mas.Length; i++)

{

if (mas[i] % 2 == 0)

{

geometric \*= mas[i];

count++;

}

}

// (x1\*x2\*x3\*x4\*x5) ^ (1/5);

result = Math.Pow(geometric, (1 / count));

}

public static void Sem2\_Lab3\_Count\_Condition(int[] mas, double result, out int index)

{

// Вывести четные числа которые больше геометрического

index = 0;

for (int i = 0; i < mas.Length; i++)

if (mas[i] > result)

index++;

}

public static void Sem2\_Lab3\_Calculate(int[] mas, double result, int index, out int[] output)

{

output = new int[index];

int k = 0;

for (int i = 0; i < mas.Length; i++)

if (mas[i] > result)

output[k++] = mas[i];

}

public static void Sem2\_Lab3\_Save\_Txt(int[] array, int[] array2)

{

StreamWriter streamWriter = File.CreateText("Массивы.txt");

streamWriter.WriteLine("Исходный массив:");

for (int i = 0; i < array.Length; i++)

{

streamWriter.WriteLine(array[i].ToString());

}

streamWriter.WriteLine("Конечный массив:");

for (int i = 0; i < array2.Length; i++)

{

streamWriter.WriteLine(array2[i].ToString());

}

streamWriter.Close();

Process.Start("Массивы.txt");

}

public static void Sem2\_Lab3\_Save\_Excel(int[] array, int[] array2)

{

Excel.Application excel = new Excel.Application();

Excel.Workbook WB = excel.Workbooks.Add();

Excel.Worksheet ws = WB.Worksheets[1];

ws.Name = "Исходный массив";

for (int i = 0; i < array.Length; i++)

{

ws.Cells[2, i + 1] = "[" + i + "]";

ws.Cells[3, i + 1] = array[i].ToString();

}

Excel.Worksheet ws2 = WB.Worksheets.Add();

ws2.Name = "Конечный массив";

for (int i = 0; i < array2.Length; i++)

{

ws2.Cells[2, i + 1] = "[" + i + "]";

ws2.Cells[3, i + 1] = array2[i].ToString();

}

excel.Visible = true;

excel.UserControl = true;

}

public static void Sem2\_Lab3\_Save\_Word(int[] array, int[] array2)

{

Word.Application word = new Word.Application();

Word.Document doc = word.Documents.Add();

Word.Range range = doc.Range();

range.Text = "Исходный массив:";

for (int i = 0; i < array.Length; i++)

{

range.Text += "[" + i + "] " + array[i].ToString();

}

range.Text += "\nКонечный массив:\n";

for (int i = 0; i < array2.Length; i++)

{

range.Text += "[" + i + "] " + array2[i].ToString();

}

word.Visible = true;

}

}

}

Основной 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;

using Dll\_lab;

using Second\_Semestr\_Dll;

using Microsoft.VisualBasic;

namespace Лабораторная\_работа

{

public partial class Sem2\_CalcLab3 : Form

{

public Sem2\_CalcLab3()

{

InitializeComponent();

}

public int[] array;

public int[] array2;

private void Make\_result\_Sem2\_lab3\_Click(object sender, EventArgs e)

{

string length\_text = Interaction.InputBox("Введите количество элементов массива для генерации", "Размер массива", "15"); // Количество

if (length\_text == "") return;

if (!int.TryParse(length\_text, out int length)) { MessageBox.Show("Неверный формат данных. Ошибка."); return; }

string min\_text = Interaction.InputBox("Нижняя граница генерации", "Минимальное значение", "-10");

if (min\_text == "") return;

if (!int.TryParse(min\_text, out int min)) { MessageBox.Show("Неверный формат данных. Ошибка."); return; }

string max\_text = Interaction.InputBox("Верхняя граница генерации", "Максимальное значение", "60");

if (max\_text == "") return;

if (!int.TryParse(max\_text, out int max)) { MessageBox.Show("Неверный формат данных. Ошибка."); return; }

// Если ввели максимальное число меньше чем меньшее

if (max < min) { MessageBox.Show("Верхняя граница генерации чисел должна быть больше нижней. Программа остановлена"); return; }

int[] mas = new int[length];

// Заполняем массив

SecSem\_Dll.Enter\_massiv(mas, length, min, max);

// Вывод исходного массива

SecSem\_Dll.Output\_massiv(mas, length, Lab2\_Sem3\_Main\_DGV);

//Вычисление среднего геометрического

SecSem\_Dll.Sem2\_Lab3\_Calc\_Geometric(mas, out double result);

//Посчитать сколько чисел подходят под условие

SecSem\_Dll.Sem2\_Lab3\_Count\_Condition(mas, result, out int index);

// Расчет и вывод массива под условие

SecSem\_Dll.Sem2\_Lab3\_Calculate(mas, result, index, out int[] output);

// Вывод для пользователя

MessageBox.Show("Найдено: " + index.ToString() + " элементов", "Результат", MessageBoxButtons.OK, MessageBoxIcon.Information);

// Вывод результата на вторую таблицу

SecSem\_Dll.Output\_massiv(output, index, Lab2\_Sem3\_Resh\_DGV);

array = mas; // Открываем массив для других функций

array2 = output;

}

private void Sem2\_Lab3\_Save\_Txt\_Click(object sender, EventArgs e)

{

SecSem\_Dll.Sem2\_Lab3\_Save\_Txt(array, array2);

}

private void Sem2\_Lab3\_Save\_Excel\_Click(object sender, EventArgs e)

{

SecSem\_Dll.Sem2\_Lab3\_Save\_Excel(array, array2);

}

private void Sem2\_Lab3\_Save\_Word\_Click(object sender, EventArgs e)

{

SecSem\_Dll.Sem2\_Lab3\_Save\_Word(array, array2);

}

private void Sem2\_Lab3\_exit\_Click(object sender, EventArgs e)

{

this.Close();

}

}

}

* 1. **Результаты работы программы**

На рисунке 7 представлено задание.

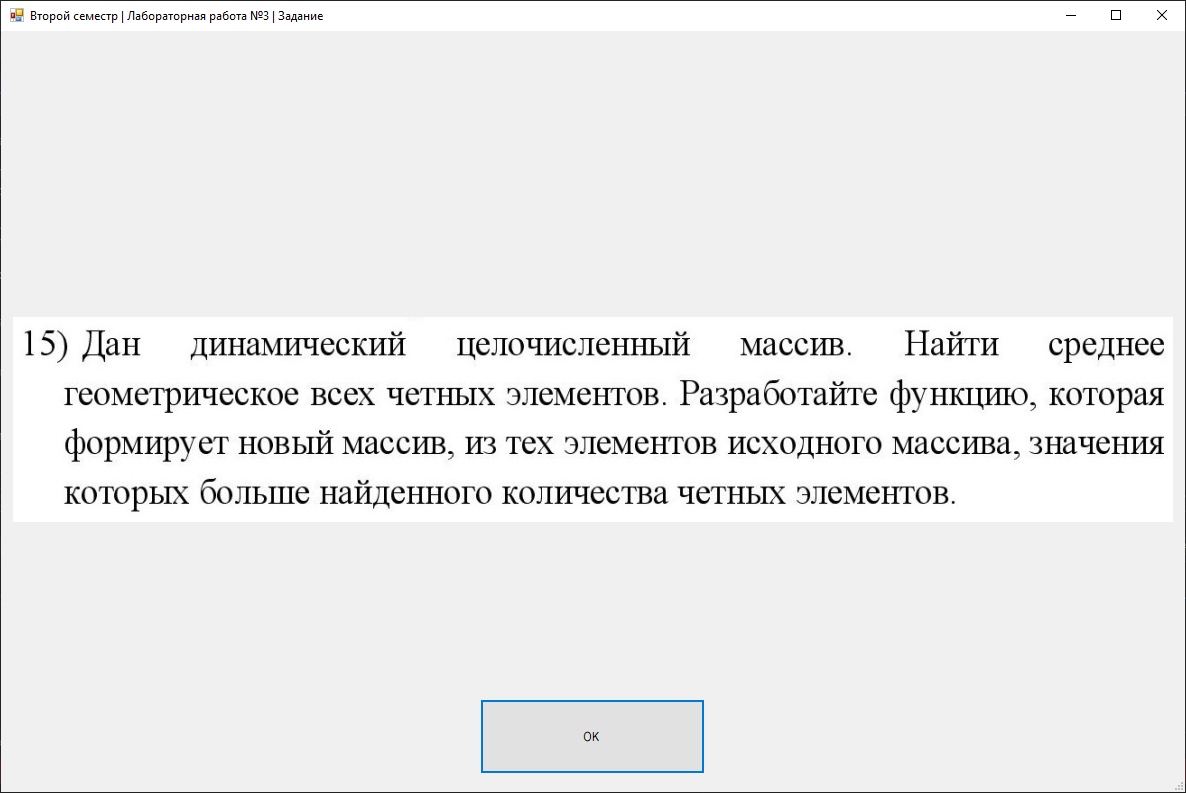


Рисунок 7 – Задание.

На рисунке 8 представлена демонстрация работы программы.

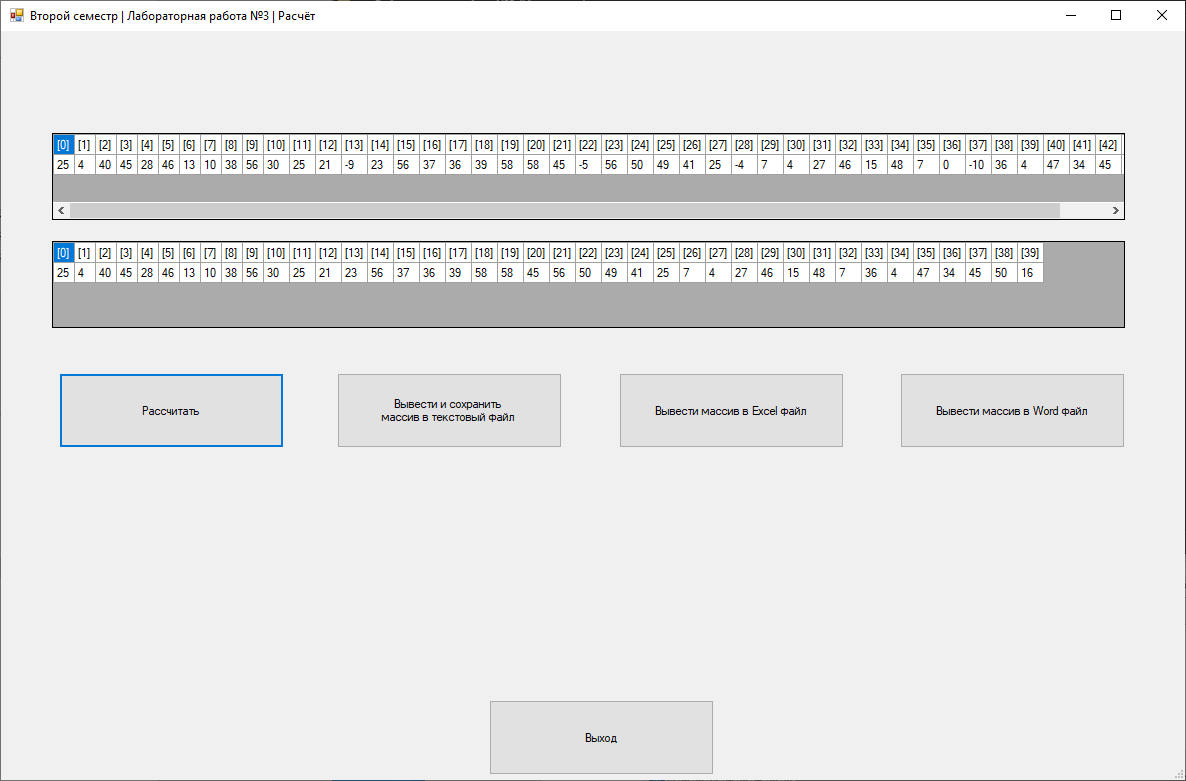


Рисунок 8 – Демонстрация работы.

На рисунке 9 представлена работа кнопки сохранения в текстовый файл.

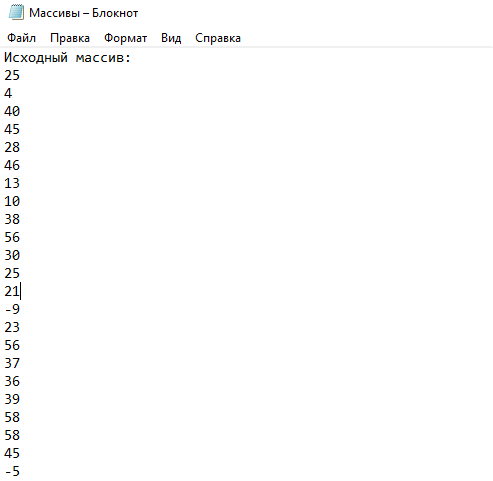


Рисунок 9 – Сохранение в текстовый файл.

На рисунке 10 представлена работа кнопки сохранения в Word файл.

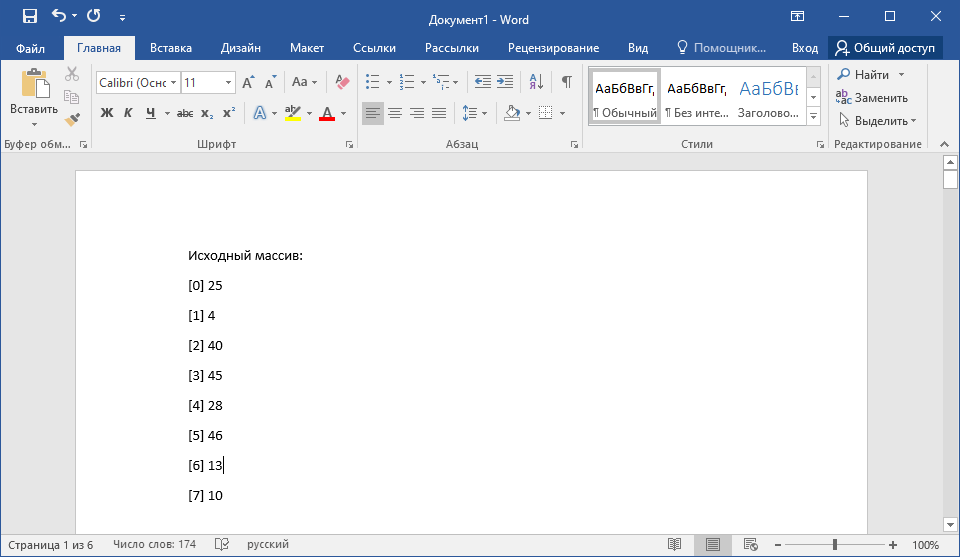


Рисунок 10 – Сохранение в Word файл.

На рисунке 11 представлена работа кнопки сохранения в Excel файл.

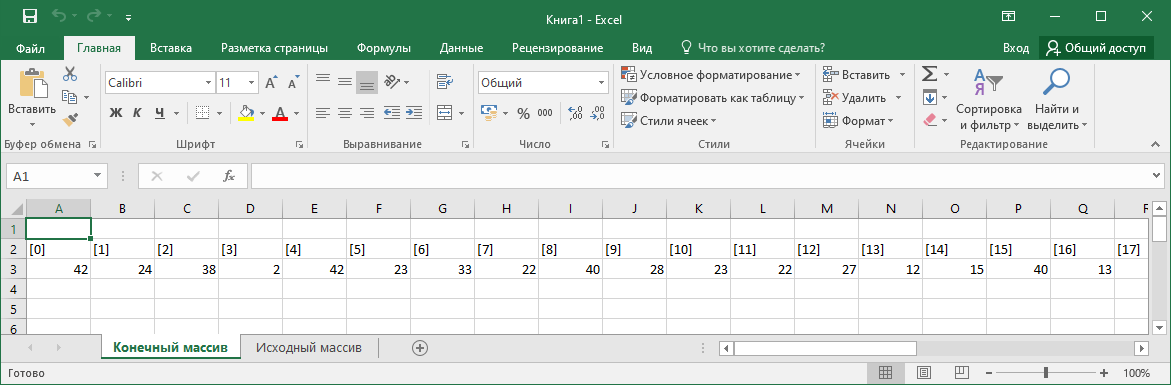


Рисунок 11 – Сохранение в Excel файл.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Гуриков, С. Р. Введение в программирование на языке Visual C#: учебное пособие / С. Р. Гуриков. — Москва: ФОРУМ: ИНФРА-М, 2019. — 447 с. — (Среднее профессиональное образование). - ISBN 978-5-00091-540-0. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1012397> (дата обращения: 03.06.2024).

2. ГОСТ 7.32-2017. Межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления" (введен в действие Приказом Росстандарта от 24.10.2017 N 1494-ст) (дата обращений 03.06.2024).