**Лабораторная работа №4**

**«Разработка класса по работе с массивами»**

**Цель работы:** работа с библиотечными модулями, разработка структуры данных и методы класса по заданию.

**Вариант №4**

**Задание 1**

1. Создать библиотечный модуль с именем ClassLibraryForArray с описанием класса IntArray по работе с одномерными целочисленными массивами. Включить в него следующие члены:

Закрытые поля:

int[] a; //закрытый одномерный массив

int length; //закрытая длина

Конструкторы:

public IntArray(int length) //конструктор 1 для создания массива заданной длины length, стр. 114-117, 158

public IntArray(params int[] arr) //конструктор 2 с переменным числом параметров, стр. 154, 163

Свойства:

public int Length //свойство Длина массива, стр. 120-123, 158

public int this[int i] // Индексатор, стр. 157-159, 164

Методы класса:

public static IntArray RandomIntArray(int length, int a, int b) // создание массива длины length и заполнение его случайными целыми числами в диапазоне от a до b, стр. 148-150

public static IntArray ArrayFromTextFile(string fileName) //ввод массива из текстового файла с именем filename, стр. 258-259

public static void ArrayToTextFile(IntArray arr, string fileName) //вывод массива arr в текстовый файл с именем filename, стр. 256-257

public static int SumArray(IntArray arr) //вычисление суммы элементов массива arr, стр. 128-129

Операции класса:

public static IntArray operator ++(IntArray arr) // ++: инкремент: увеличение на 1 всех элементов массива, 163-164

public static IntArray operator + (IntArray x, int y) // +: сложение массива x со скаляром y, стр. 165-167

public static IntArray operator +(int x, IntArray y) // +: сложение скаляра x с массивом y, стр. 165-167

public static IntArray operator +(IntArray x, IntArray y) // +: сложение двух массивов x и y, стр. 165-167

public static IntArray operator --(IntArray arr) // --: декремент: уменьшение на 1 всех элементов массива

public static IntArray operator -(IntArray x, int y) // -: вычитание из массива x скаляра y (x - y)

public static IntArray operator -(int x, IntArray y) // -: вычитание из скаляра x массива y (x - y)

public static IntArray operator -(IntArray x, IntArray y) // -: вычитание из массива x массива y (x - y)

1. Описать реализацию всех функциональных членов.
2. Для тестирования библиотечного модуля создать консольное приложение (проект) с именем ConsoleAppTest. Подключите к нему ваш модуль. Для подключения модуля необходимо добавить к проекту ссылку на этот модуль (Project\Add Reference\Browse) и указать имя модуля в списке используемых модулей с помощью директивы: using <имя модуля>. Для информирования пользователя о том, какой метод отработал, добавить обработчик события Notify.

На рисунках 1-6 изображены код для создания библиотечного модуля с именем ClassLibraryForArray

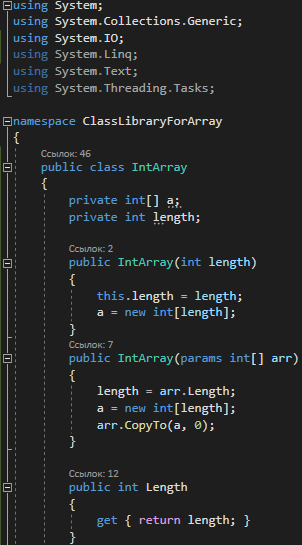


Рис. 1

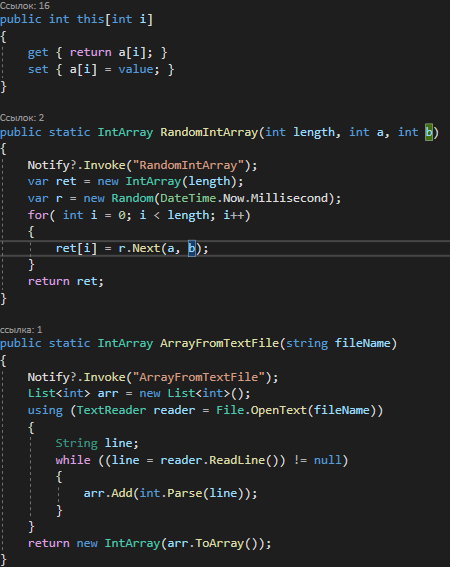


Рис. 2

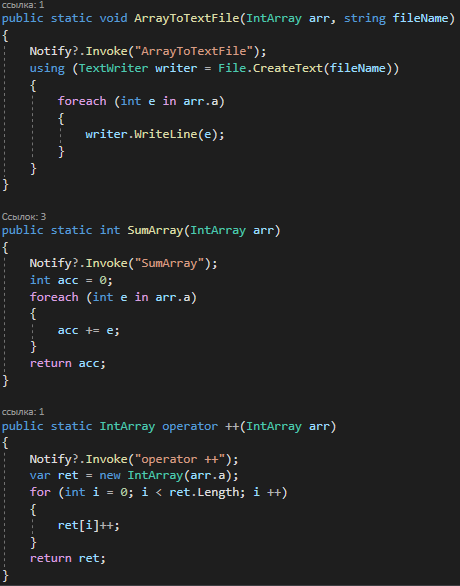


Рис. 3

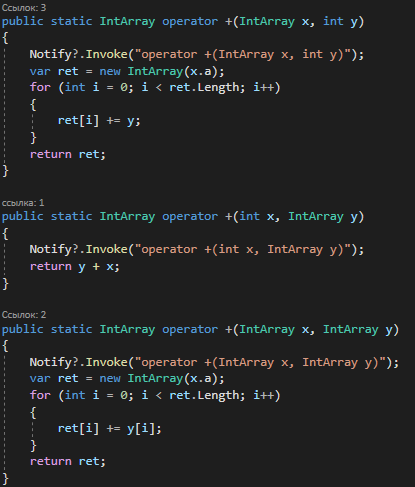


Рис. 4

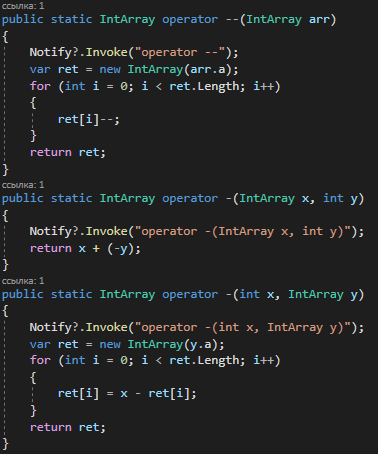


Рис. 5

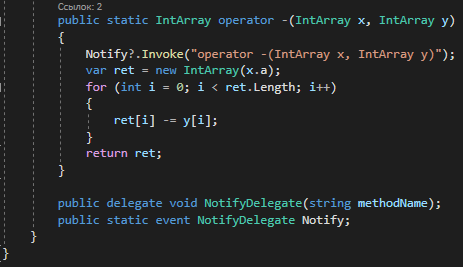


Рис. 6

На рисунке 7 изображён код для выведения результатов библиотечного модуля

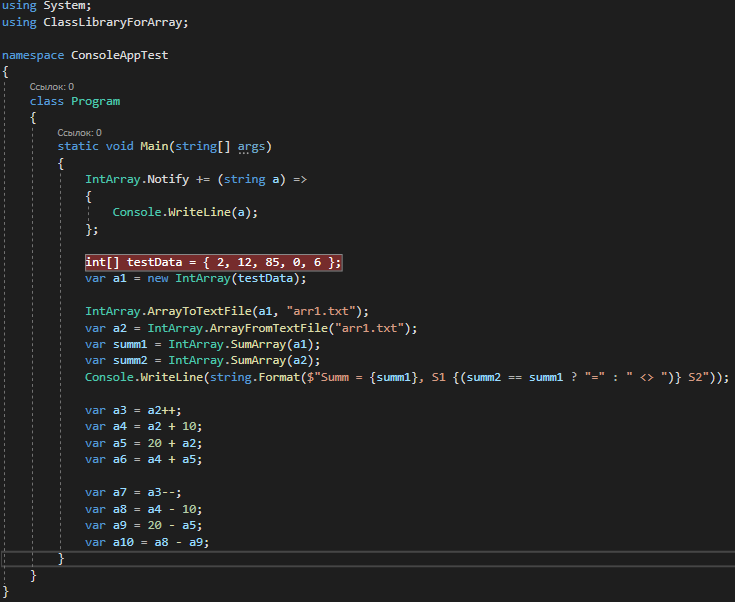


Рис. 7

На рисунке 8 изображён результат закрытых полей

Изображение выглядит как снимок экрана, линия, текст

Автоматически созданное описание

Рис. 8

На рисунках 9-10 изображены результат ввода массива из текстового файла с именем filename

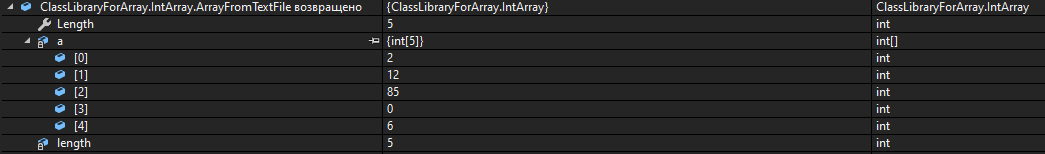


Рис. 9

Изображение выглядит как текст, снимок экрана, Шрифт, дисплей

Автоматически созданное описание

Рис. 10

На рисунке 11 изображены результаты суммирования элементов массива



Рис. 11

На рисунке 12 изображён результат сложения всех элементов массива на единицу

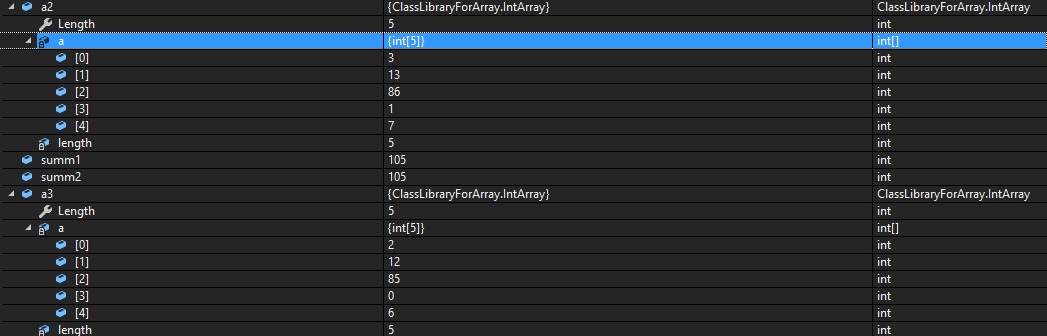


Рис. 12

На рисунке 13 изображён результат сложения элементов массива на 10

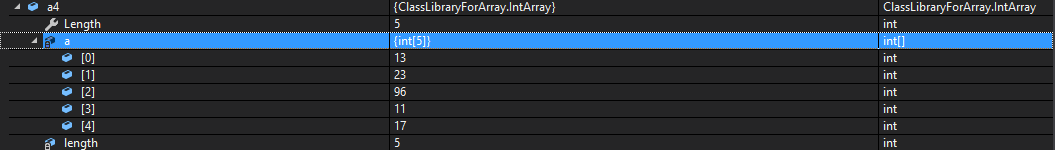


Рис. 13

На рисунке 14 изображён результат сложения элементов массива на 20

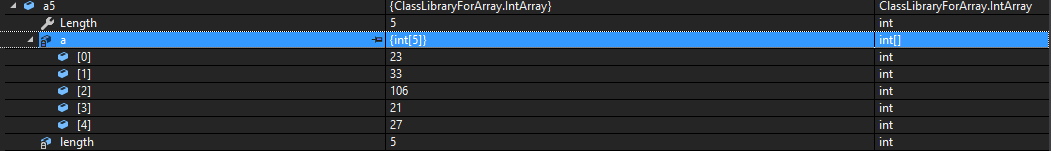


Рис. 14

На рисунке 15 изображён результат сложения элементов массива друг на друга

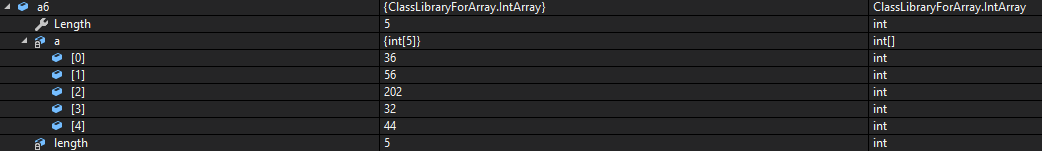


Рис. 15

На рисунке 16 изображён результат вычитания элементов массива на единицу

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рис. 16

На рисунке 17 изображён результат вычитания элементов массива на 10

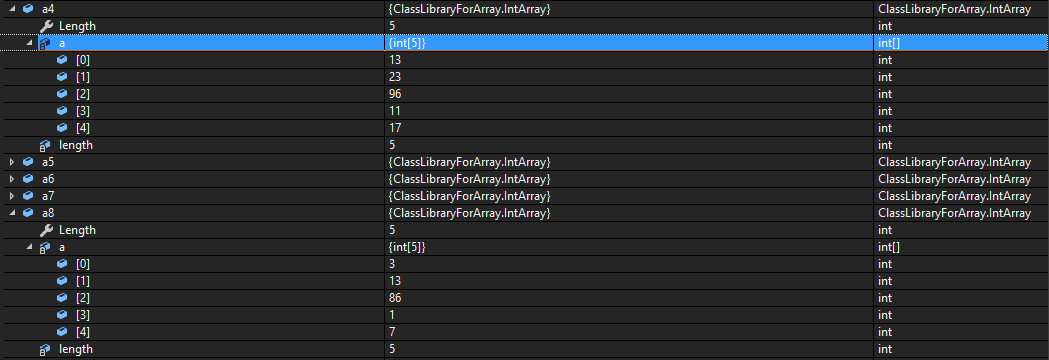


Рис. 17

Текст программы библиотечного модуля:

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ClassLibraryForArray

{

public class IntArray

{

private int[] a;

private int length;

public IntArray(int length)

{

this.length = length;

a = new int[length];

}

public IntArray(params int[] arr)

{

length = arr.Length;

a = new int[length];

arr.CopyTo(a, 0);

}

public int Length

{

get { return length; }

}

public int this[int i]

{

get { return a[i]; }

set { a[i] = value; }

}

public static IntArray RandomIntArray(int length, int a, int b)

{

Notify?.Invoke("RandomIntArray");

var ret = new IntArray(length);

var r = new Random(DateTime.Now.Millisecond);

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

{

ret[i] = r.Next(a, b);

}

return ret;

}

public static IntArray ArrayFromTextFile(string fileName)

{

Notify?.Invoke("ArrayFromTextFile");

List<int> arr = new List<int>();

using (TextReader reader = File.OpenText(fileName))

{

String line;

while ((line = reader.ReadLine()) != null)

{

arr.Add(int.Parse(line));

}

}

return new IntArray(arr.ToArray());

}

public static void ArrayToTextFile(IntArray arr, string fileName)

{

Notify?.Invoke("ArrayToTextFile");

using (TextWriter writer = File.CreateText(fileName))

{

foreach (int e in arr.a)

{

writer.WriteLine(e);

}

}

}

public static int SumArray(IntArray arr)

{

Notify?.Invoke("SumArray");

int acc = 0;

foreach (int e in arr.a)

{

acc += e;

}

return acc;

}

public static IntArray operator ++(IntArray arr)

{

Notify?.Invoke("operator ++");

var ret = new IntArray(arr.a);

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

{

ret[i]++;

}

return ret;

}

public static IntArray operator +(IntArray x, int y)

{

Notify?.Invoke("operator +(IntArray x, int y)");

var ret = new IntArray(x.a);

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

{

ret[i] += y;

}

return ret;

}

public static IntArray operator +(int x, IntArray y)

{

Notify?.Invoke("operator +(int x, IntArray y)");

return y + x;

}

public static IntArray operator +(IntArray x, IntArray y)

{

Notify?.Invoke("operator +(IntArray x, IntArray y)");

var newSize = Math.Min(x.Length, y.Length);

var ret = new IntArray(newSize);

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

{

ret[i] = x[i] + y[i];

}

return ret;

}

public static IntArray operator --(IntArray arr)

{

Notify?.Invoke("operator --");

var ret = new IntArray(arr.a);

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

{

ret[i]--;

}

return ret;

}

public static IntArray operator -(IntArray x, int y)

{

Notify?.Invoke("operator -(IntArray x, int y)");

return x + (-y);

}

public static IntArray operator -(int x, IntArray y)

{

Notify?.Invoke("operator -(int x, IntArray y)");

var ret = new IntArray(y.a);

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

{

ret[i] = x - ret[i];

}

return ret;

}

public static IntArray operator -(IntArray x, IntArray y)

{

Notify?.Invoke("operator -(IntArray x, IntArray y)");

var newSize = Math.Min(x.Length, y.Length);

var ret = new IntArray(newSize);

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

{

ret[i] = x[i] - y[i];

}

return ret;

}

public delegate void NotifyDelegate(string methodName);

public static event NotifyDelegate Notify;

}

}

Текст программы для проверки работы:

using System;

using ClassLibraryForArray;

namespace ConsoleAppTest

{

class Program

{

static void Main(string[] args)

{

IntArray.Notify += (string a) =>

{

Console.WriteLine(a);

};

int[] testData = { 2, 12, 85, 0, 6 };

var a1 = new IntArray(testData);

IntArray.ArrayToTextFile(a1, "arr1.txt");

var a2 = IntArray.ArrayFromTextFile("arr1.txt");

var summ1 = IntArray.SumArray(a1);

var summ2 = IntArray.SumArray(a2);

Console.WriteLine(string.Format($"Summ = {summ1}, S1 {(summ2 == summ1 ? "=" : " <> ")} S2"));

var a3 = a2++;

var a4 = a2 + 10;

var a5 = 20 + a2;

var a6 = a4 + a5;

var a7 = a3--;

var a8 = a4 - 10;

var a9 = 20 - a5;

var a10 = a8 - a9;

}

}

}

**Задание 2 (по вариантам)**

Создать приложение Windows Form с графическим интерфейсом. Подключить к нему созданный в 1-й части задания библиотечный модуль. Основная программа должна обрабатывать 2 массива разной длины. Длина каждого массива запрашивается у пользователя. Для работы с массивами надо предусмотреть на форме 2 таблицы (компонент DataGridVew), размеры которых задаются в процессе выполнения программы. Заполнение каждого массива может происходить при помощи генератора случайных чисел, вручную или из текстового файла (в последнем случае размер массива должен определяться по количеству чисел в файле). Для выбора режима заполнения массива используйте переключатель «зависимая кнопка выбора».

Для каждого массива вычисляется: сумма элементов, количество элементов, кратных заданному числу, а также выполняется сложение и вычитание двух массивов. Результат сложения или вычитания вывести в дополнительную таблицу. Добавить функцию для обработки вашего варианта. Пользовательский интерфейс может быть оформлен в виде системы меню (MenuStrip) или панели с кнопками (панели инструментов ToolStrip).

**Вариант №4**

Разработать метод, который в массиве из n целых чисел подсчитывает количество отрицательных и положительных элементов, и если отрицательных элементов окажется больше, меняет знаки всех элементов на противоположные.

На рисунке 18 изображена основная форма для работы

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рис. 18

На рисунках 19-26 изображены основной код для работы над формой

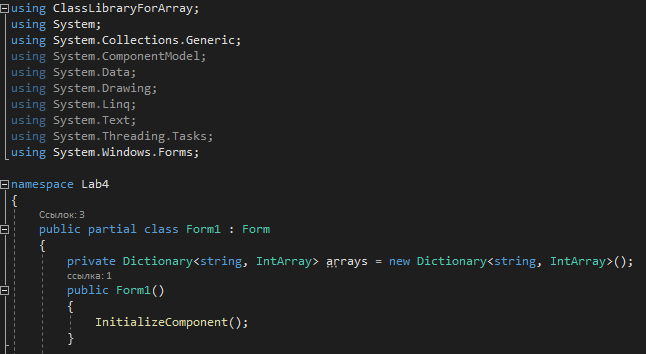


Рис. 19

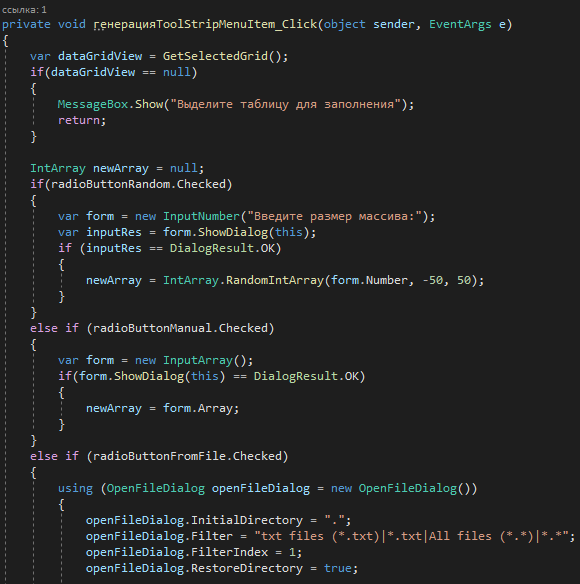


Рис. 20

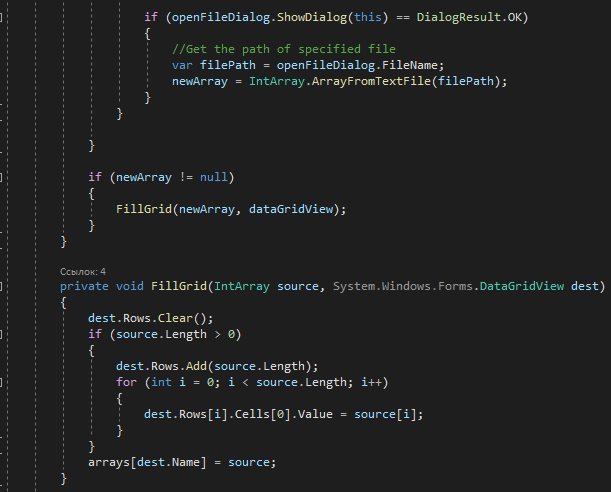


Рис. 21



Рис. 22



Рис. 23

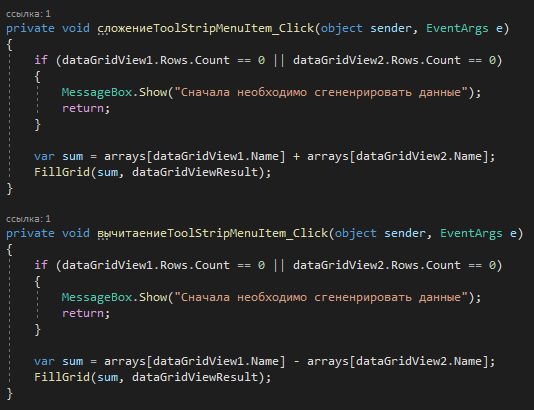


Рис. 24

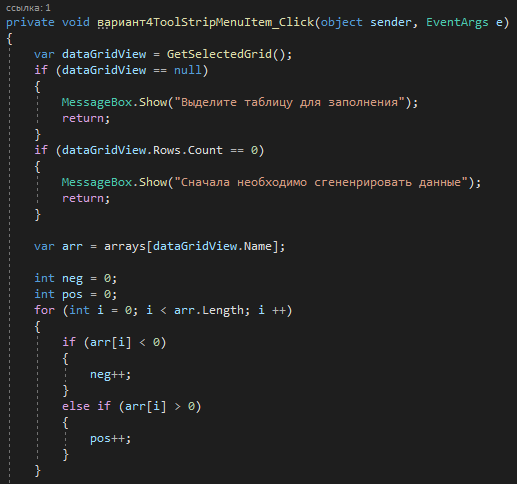


Рис. 25



Рис. 26

На рисунках 27-28 изображены код для работы кнопок

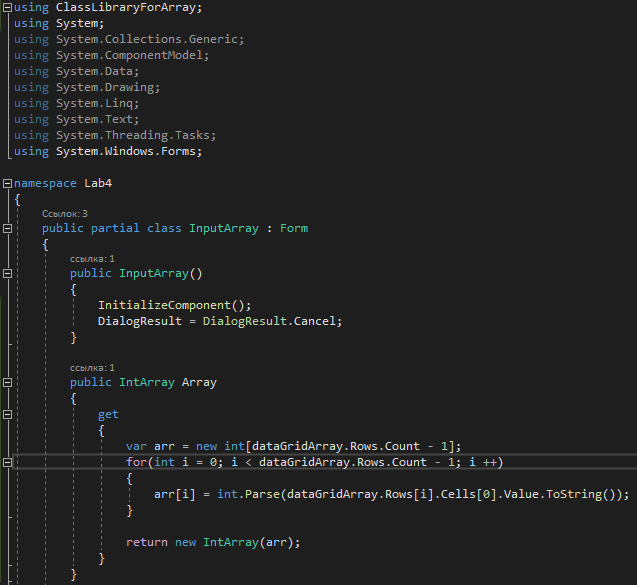


Рис. 27

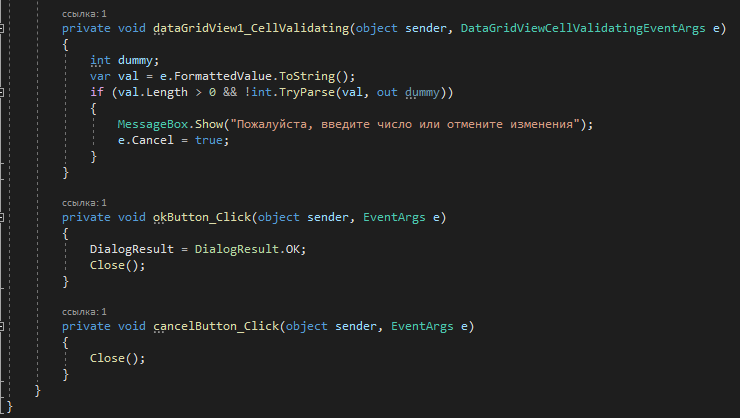


Рис. 28

На рисунке 29 изображён код для работы формы при выборе «из файла»

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Рис. 29

На рисунке 30 изображена генерация массивов после запуска кода когда выбран способ генерации массива «Генератор случайных чисел»

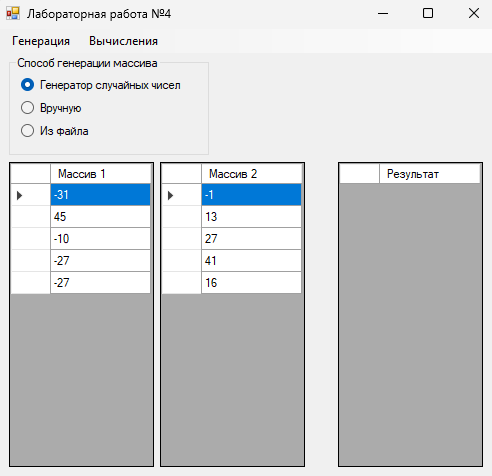


Рис. 30

На рисунке 31 изображён результат вычисления «Сумма всех элементов»

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рис. 31

На рисунке 32 изображён результат вычисления «Количество кратных»

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рис. 32

На рисунке 33 изображён результат вычисления «Сложение массивов»

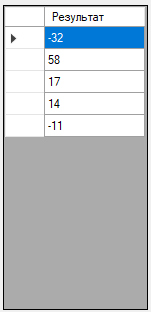


Рис. 33

На рисунке 34 изображён результат вычисления «Вычитание массивов»

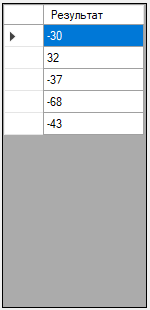


Рис. 34

На рисунке 35 изображён результат преобразования отрицательных чисел в положительные, а положительные – в отрицательные.

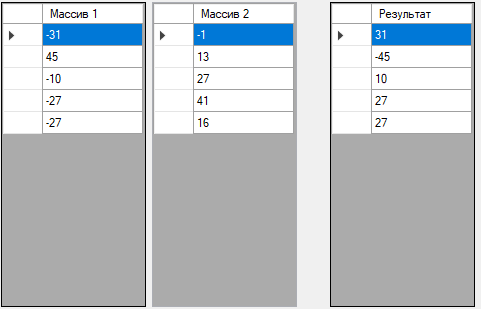


Рис. 35

На рисунке 36 изображён результат результат преобразования отрицательных чисел в положительные, а положительные – в отрицательные, если отрицательных чисел мало

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рис. 36

На рисунке 37 изображена форма для ручного заполнения массива

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рис. 37

На рисунке 38 изображена форма, заполненная с клавиатуры

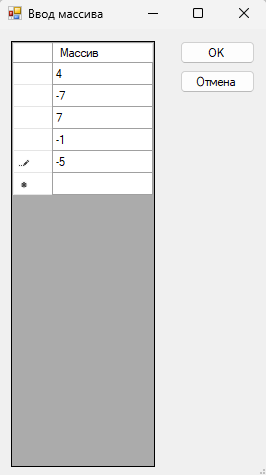


Рис. 38

На рисунке 39 изображено открытие файла, если мы выбираем способ генерации массива «Из файла»

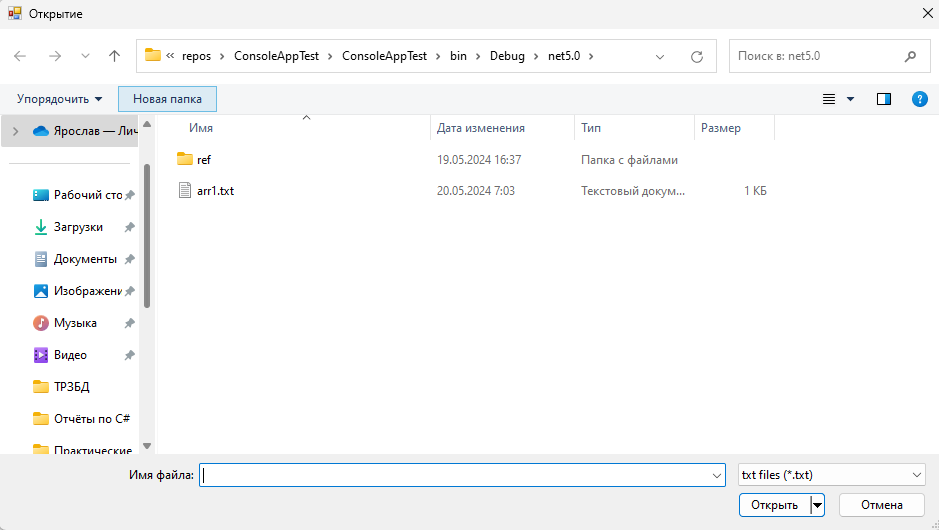


Рис. 39

На рисунке 40 изображён результат

Изображение выглядит как снимок экрана, текст, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рис. 40