**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3

по дисциплине «Объектно-ориентированное проектирование и программирование»

на тему: «Операции и перегруженные методы класса»

Выполнил: студент гр. ИТП-21

Трацевский И. С.

Принял: преподаватель

Гуменников Е. Д.

Гомель 2022

**Цель работы:** освоить создание перегруженных методов класса.

**Задание:** Создать класс «Одномерный массив», в котором описать следующие элементы:

• закрытое поле – массив вещественных чисел;

• свойство для определения длины массива;

• индексатор для доступа к элементам поля-массива;

• конструктор с параметрами;

• методы ввода и вывода массива;

• перегруженные методы для определения количества отрицательных элементов во всем массиве, определения количества элементов расположенных после элемента с заданным номером, а также для определения количества отрицательных элементов, больших заданного числа;

• операция поэлементного умножения массивов (если массивы разной размерности, в результирующий массив оставшиеся элементы большего массива добавляются без изменения); • операция неявного преобразования целого числа в объект класса «Одномерный массив» (результатом является массив, размерность которого равна преобразуемому числу, заполненный единицами).

Разработать программу, выполняющую следующие действия:

− Ввод и вывод трех массивов A, В и С;

− Вычисление количества отрицательных элементов массива А\*С;

− Вычисление количества отрицательных элементов в массивах A и С, расположенных после элемента с заданным номером;

**Ход работы:**

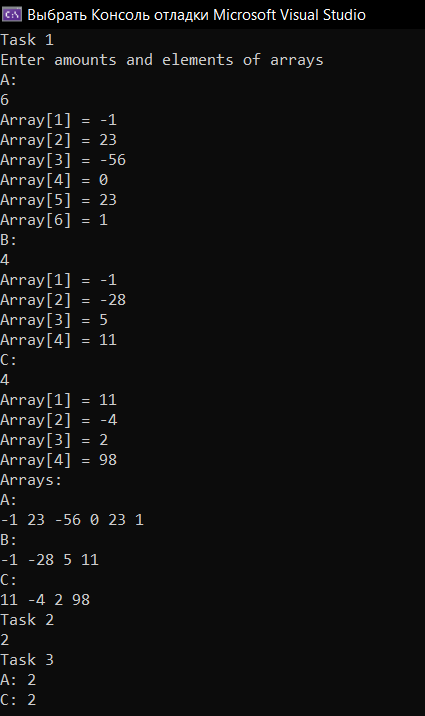


Рисунок 1 – Результат выполнения программы

**Вывод:** освоены перегрузки методов.

**Листинг:**

**Файл класса:**

public class SingleArr

{

private double[] array = { };

public SingleArr(double[] arr)

{

this.array = arr;

}

public SingleArr()

{

}

public double[] getArr()

{

return this.array;

}

public int getLength()

{

return array.Length;

}

public double getAnEllement(int index)

{

return array[index];

}

public void printArr()

{

Console.WriteLine(string.Join(" ", this.array));

}

public void enterArr()

{

int arrSize = Convert.ToInt32(Console.ReadLine());

double[] arr = new double[arrSize];

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

{

Console.Write($"Array[{i + 1}] = ");

arr[i] = Convert.ToDouble(Console.ReadLine());

}

this.array = arr;

}

public int getAmountOfNegative()

{

return this.array.Where(elem => elem < 0).Count();

}

public int getAmountOfNegative(int afterInd)

{

int counter = 0;

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

{

if (this.array[i] < 0)

{

counter++;

}

}

return counter;

}

public int getAmountOfNegative(double allAboveNumb)

{

return this.array.Where(elem => elem < 0 && elem > allAboveNumb).Count();

}

static public int[] convertToArr(int newArrSize)

{

int[] arr = new int[newArrSize];

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

{

arr[i] = 1;

}

return arr;

}

static public double[] multiplication(double[] arr1, double[] arr2)

{

int length1 = arr1.Length;

int length2 = arr2.Length;

double[] remainArr = {};

int lengthOfMin = length1;

int lengthOfMax = length1;

if (length1 > length2)

{

lengthOfMin = length2;

remainArr = new Double[length1 - length2];

Array.Copy(arr1, length2, remainArr, 0, length1 - length2);

} else if(length2 > length1)

{

lengthOfMax = length2;

remainArr = new Double[length2 - length1];

Array.Copy(arr2, length1, remainArr, 0, length2 - length1);

}

double[] multipliedArr = new double[lengthOfMin];

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

{

multipliedArr[i] = arr1[i] \* arr2[i];

}

double[] resultArr = multipliedArr.Concat(remainArr).ToArray();

return resultArr;

}

}

**Файл программы:**

using customArrsLib;

Console.WriteLine("Task 1");

SingleArr A = new SingleArr();

SingleArr B = new SingleArr();

SingleArr C = new SingleArr();

Console.WriteLine("Enter amounts and elements of arrays");

Console.WriteLine("A:");

A.enterArr();

Console.WriteLine("B:");

B.enterArr();

Console.WriteLine("C:");

C.enterArr();

Console.WriteLine("Arrays:");

Console.WriteLine("A:");

A.printArr();

Console.WriteLine("B:");

B.printArr();

Console.WriteLine("C:");

C.printArr();

Console.WriteLine("Task 2");

Console.WriteLine(new SingleArr(SingleArr.multiplication(A.getArr(), B.getArr())).getAmountOfNegative());

Console.WriteLine("Task 3");

double lookAfterNumb = -100.0;

Console.WriteLine("A: " + A.getAmountOfNegative(lookAfterNumb));

Console.WriteLine("C: " + B.getAmountOfNegative(lookAfterNumb));