Министерство образования Республики Беларусь

Оршанский колледж ВГУ имени П.М.Машерова

**Отчёт**

по учебной практике по программированию

«Модуль 1.2»

Выполнил учащийся Красантович З.К.

группы 3ПОИС23 30.09.2025г.

Проверил Алейников М.А.

30.09.2025г.

Орша, 2025

**Цель:** выполнить задания модулей 1.2.

**Содержание работы.**

**Задача 1.**

Ввести размер массива N и значения его элементов. Нормировать элементы массива, разделив их на значение максимального по модулю элемента. Вывести значения элементов изменённого массива.

**Задача 2.**

Определить и инициализировать целочисленный массив из 10-ти элементов. Ввести целое число и заменить им значение максимального элемента в массиве.

**Задача 3.**

Вычислить К простым числам. Значение К ввести с клавиатуры. Вывести значения чисел, размещая их по 10 на строке.

**Задача 4.**

Определить целочисленный массив из К элементов. Присвоить элементам случайные значения из диапазона [А, В). Найти индексы минимального и максимального элементов массива. Вывести значения элементов, расположенных между найденными (включая найденные).

**Задача 5.**

Определить символьный массив из К элементов. Присвоить элементам случайные значения букв русского алфавита. Создать новый массив, поместив в него только согласные буквы из первого массива. Значение К ввести с клавиатуры. Вывести элементы обоих массивов.

**Задача 6.**

Определить вещественный массив из 10-ти элементов. Присвоить элементам случайные значения из диапазона [–10, 10). Сформировать массив индексов, которые нумеруют элементы первого массива в порядке возрастания их значений.

**Ход выполнения работы.**

**Задача 1.**

Ввести размер массива N и значения его элементов. Нормировать элементы массива, разделив их на значение максимального по модулю элемента. Вывести значения элементов изменённого массива.

Для данной задачи необходимо написать программу, которая будет запрашивать размер массива и после чего предлагать заполнить его.

В листинге 1 представлена программа для выполнения задания 1.

Листинг 1. Код программы задания 1.

using System;

namespace Задание\_1.\_2.\_1

{

class Program

{

static void Main()

{

Console.WriteLine("Нормировка массива");

Console.WriteLine("Программа делит все элементы массива на максимальный по модулю элемент");

while (true)

{

Console.Write("\nВведите размер массива N (или 'q' для выхода): ");

string input = Console.ReadLine();

if (input.ToLower() == "q")

{

Console.WriteLine("Программа завершена. До свидания!");

break;

}

if (int.TryParse(input, out int n) && n > 0)

{

double[] array = new double[n];

Console.WriteLine($"Введите {n} элементов массива:");

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

{

Console.Write($"Элемент [{i}]: ");

if (double.TryParse(Console.ReadLine(), out double value))

array[i] = value;

else

{

Console.WriteLine("Ошибка: введено не число. Повторите ввод.");

i--;

}

}

Console.WriteLine("\nИсходный массив:");

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

{

Console.Write($"{array[i]:F4} ");

if ((i + 1) % 5 == 0) Console.WriteLine();

}

Console.WriteLine();

double maxAbs = 0;

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

{

double absValue = Math.Abs(array[i]);

if (absValue > maxAbs) maxAbs = absValue;

}

if (maxAbs == 0)

{

Console.WriteLine("Максимальный по модулю элемент равен 0. Нормировка невозможна.");

}

else

{

Console.WriteLine($"Максимальный по модулю элемент: {maxAbs:F4}");

Console.WriteLine("Нормированный массив:");

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

{

Console.Write($"{(array[i] / maxAbs):F6} ");

if ((i + 1) % 5 == 0) Console.WriteLine();

}

Console.WriteLine();

}

}

else

{

Console.WriteLine("Ошибка: размер массива должен быть положительным целым числом.");

}

}

Console.WriteLine("\nНажмите любую клавишу для выхода...");

Console.ReadKey();

}

}

}

На рисунке 1 представлен результат работы программы.

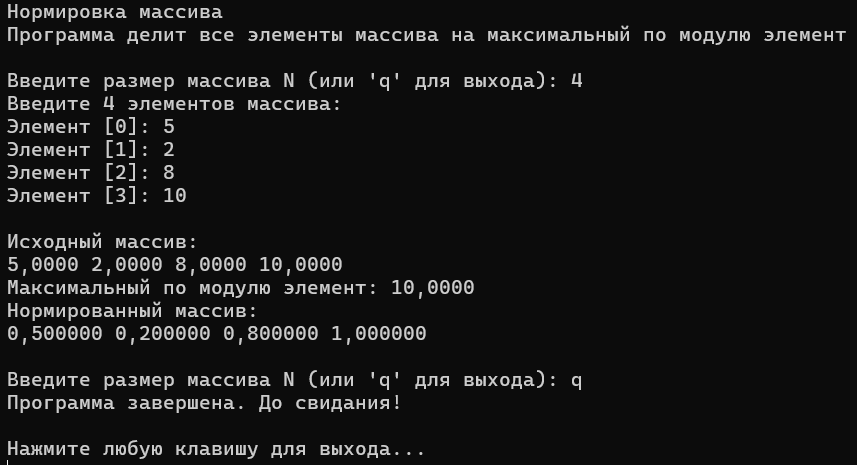


Рисунок 1 – Результат работы программы

**Задача 2.**

Определить и инициализировать целочисленный массив из 10-ти элементов. Ввести целое число и заменить им значение максимального элемента в массиве.

В листинге 2 представлен результат работы программы.

Листинг 2. Код программы задания 2.

using System;

namespace Задание\_1.\_2.\_2

{

class Program

{

static void Main()

{

// Инициализация массива (с несколькими одинаковыми максимальными значениями)

int[] array = { 5, 23, 8, 23, 15, 7, 18, 3, 21, 23 };

Console.WriteLine("Исходный массив:");

PrintArray(array);

// Находим максимальное значение

int maxValue = array[0];

for (int i = 1; i < array.Length; i++)

{

if (array[i] > maxValue)

{

maxValue = array[i];

}

}

Console.WriteLine($"\nМаксимальный элемент: {maxValue}");

// Ввод нового значения

Console.Write("\nВведите целое число для замены максимального элемента: ");

string input = Console.ReadLine();

if (int.TryParse(input, out int newValue))

{

// Заменяем все максимальные элементы

int count = 0;

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

{

if (array[i] == maxValue)

{

array[i] = newValue;

count++;

}

}

Console.WriteLine("\nМассив после замены:");

PrintArray(array);

Console.WriteLine($"Заменено {count} максимальных элементов на значение: {newValue}");

}

else

{

Console.WriteLine("Ошибка: введено не целое число.");

}

Console.WriteLine("\nНажмите любую клавишу для выхода...");

Console.ReadKey();

}

// Метод для вывода массива

static void PrintArray(int[] array)

{

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

{

Console.Write($"{array[i],3} ");

}

Console.WriteLine();

}

}

}

На рисунке 2 представлен результат работы программы.

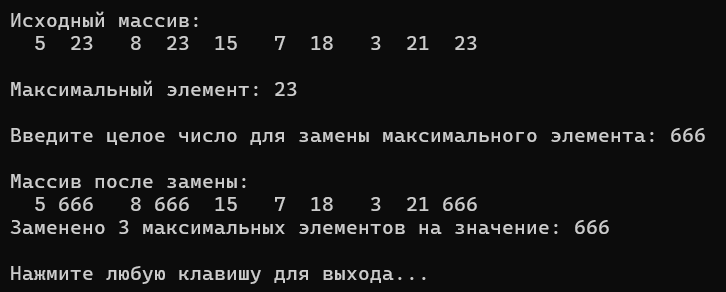


Рисунок 2 – Результат работы программы

**Задача 3.**

Вычислить К простым числам. Значение К ввести с клавиатуры. Вывести значения чисел, размещая их по 10 на строке.

Код выполненной программы представлен в листинге 3.

Листинг 3. Код программы задания 3.

using System;

using System.Collections.Generic;

namespace Задание\_1.\_2.\_3

{

class Program

{

static void Main()

{

Console.WriteLine("Программа для вычисления K простых чисел");

while (true)

{

Console.Write("\nВведите количество простых чисел K (или 'q' для выхода): ");

string input = Console.ReadLine();

if (input.ToLower() == "q")

{

Console.WriteLine("Программа завершена. До свидания!");

break;

}

if (int.TryParse(input, out int k) && k > 0)

{

List<int> primes = GeneratePrimes(k);

Console.WriteLine($"\nПервые {k} простых чисел:");

PrintPrimes(primes);

}

else

{

Console.WriteLine("Ошибка: введите положительное целое число.");

}

}

Console.WriteLine("\nНажмите любую клавишу для выхода...");

Console.ReadKey();

}

// Метод для генерации K простых чисел

static List<int> GeneratePrimes(int k)

{

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

if (k <= 0) return primes;

// Первое простое число

primes.Add(2);

// Если нужно только одно простое число

if (k == 1) return primes;

int number = 3; // Начинаем проверять с 3

while (primes.Count < k)

{

if (IsPrime(number))

{

primes.Add(number);

}

number += 2; // Проверяем только нечетные числа (кроме 2)

}

return primes;

}

// Метод для проверки, является ли число простым

static bool IsPrime(int n)

{

if (n < 2) return false;

if (n == 2) return true;

if (n % 2 == 0) return false;

// Проверяем делимость на нечетные числа до квадратного корня из n

int limit = (int)Math.Sqrt(n);

for (int i = 3; i <= limit; i += 2)

{

if (n % i == 0)

return false;

}

return true;

}

// Метод для вывода простых чисел по 10 на строку

static void PrintPrimes(List<int> primes)

{

for (int i = 0; i < primes.Count; i++)

{

Console.Write($"{primes[i],6}");

// Переход на новую строку после каждых 10 чисел

if ((i + 1) % 10 == 0)

Console.WriteLine();

}

// Если последняя строка не полная, переводим строку

if (primes.Count % 10 != 0)

Console.WriteLine();

}

}

}

На рисунке 3 представлен результат работы программы.

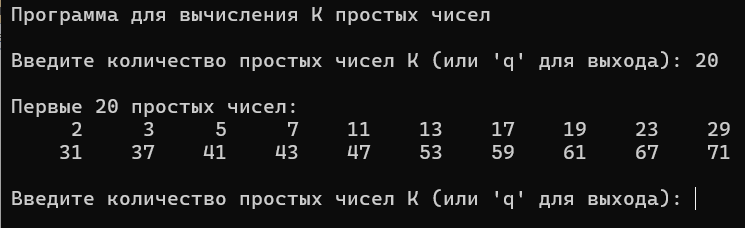


Рисунок 3 – Результат работы программы

**Задача 4.**

Определить целочисленный массив из К элементов. Присвоить элементам случайные значения из диапазона [А, В). Найти индексы минимального и максимального элементов массива. Вывести значения элементов, расположенных между найденными (включая найденные).

Ниже в листинге представлена программа, которая

Листинг 4. Код программы задания 4.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Задание\_1.\_2.\_4

{

class Program

{

static void Main()

{

Console.WriteLine("Программа для работы с массивом");

Console.WriteLine("Поиск минимального и максимального элементов и вывод значений между ними");

while (true)

{

Console.Write("\nВведите количество элементов K (или 'q' для выхода): ");

string input = Console.ReadLine();

if (input.ToLower() == "q")

{

Console.WriteLine("Программа завершена. До свидания!");

break;

}

if (int.TryParse(input, out int k) && k > 0)

{

Console.Write("Введите нижнюю границу A: ");

if (!int.TryParse(Console.ReadLine(), out int a))

{

Console.WriteLine("Ошибка: введите целое число для A.");

continue;

}

Console.Write("Введите верхнюю границу B: ");

if (!int.TryParse(Console.ReadLine(), out int b))

{

Console.WriteLine("Ошибка: введите целое число для B.");

continue;

}

if (b <= a)

{

Console.WriteLine("Ошибка: B должно быть больше A.");

continue;

}

// Создаем и заполняем массив

int[] array = CreateRandomArray(k, a, b);

// Выводим исходный массив

Console.WriteLine("\nИсходный массив:");

PrintArray(array);

// Находим индексы минимального и максимального элементов

FindMinMaxIndices(array, out int minIndex, out int maxIndex);

// Выводим элементы между minIndex и maxIndex (включая их)

Console.WriteLine("\nЭлементы между минимальным и максимальным (включая их):");

PrintRange(array, minIndex, maxIndex);

}

else

{

Console.WriteLine("Ошибка: K должно быть положительным целым числом.");

}

}

Console.WriteLine("\nНажмите любую клавишу для выхода...");

Console.ReadKey();

}

// Метод для создания массива со случайными числами в диапазоне [A, B)

static int[] CreateRandomArray(int k, int a, int b)

{

Random random = new Random();

int[] array = new int[k];

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

{

array[i] = random.Next(a, b); // [A, B) - включая A, не включая B

}

return array;

}

// Метод для поиска индексов минимального и максимального элементов

static void FindMinMaxIndices(int[] array, out int minIndex, out int maxIndex)

{

minIndex = 0;

maxIndex = 0;

for (int i = 1; i < array.Length; i++)

{

if (array[i] < array[minIndex])

{

minIndex = i;

}

if (array[i] > array[maxIndex])

{

maxIndex = i;

}

}

Console.WriteLine($"\nМинимальный элемент: array[{minIndex}] = {array[minIndex]}");

Console.WriteLine($"Максимальный элемент: array[{maxIndex}] = {array[maxIndex]}");

}

// Метод для вывода элементов массива между двумя индексами (включая их)

static void PrintRange(int[] array, int index1, int index2)

{

// Определяем начальный и конечный индексы

int start = Math.Min(index1, index2);

int end = Math.Max(index1, index2);

Console.Write($"Индексы от {start} до {end}: ");

for (int i = start; i <= end; i++)

{

Console.Write($"{array[i]} ");

}

Console.WriteLine();

// Дополнительная информация

Console.WriteLine($"Количество элементов в диапазоне: {end - start + 1}");

}

// Метод для вывода всего массива

static void PrintArray(int[] array)

{

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

{

Console.Write($"{array[i],4}");

if ((i + 1) % 10 == 0)

Console.WriteLine();

}

if (array.Length % 10 != 0)

Console.WriteLine();

}

}

}

На рисунке 4 представлен результат работы программы.

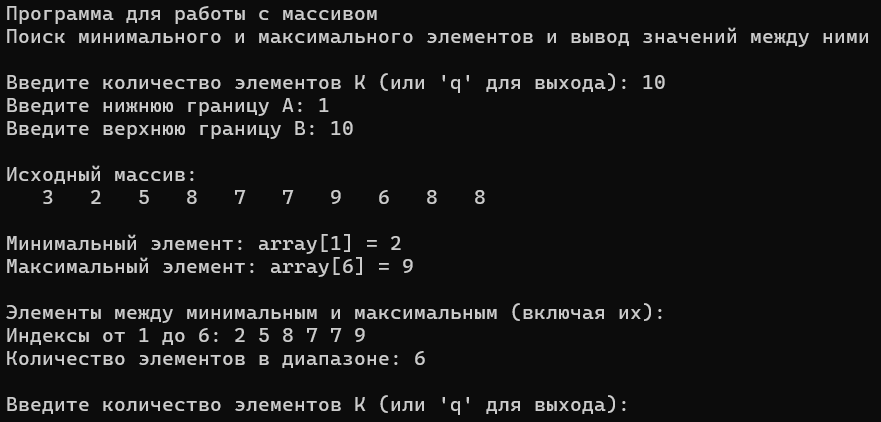


Рисунок 4 – Результат работы программы

**Задача 5.**

Определить символьный массив из К элементов. Присвоить элементам случайные значения букв русского алфавита. Создать новый массив, поместив в него только согласные буквы из первого массива. Значение К ввести с клавиатуры. Вывести элементы обоих массивов.

В листинге 5 представлен код программы к заданию 5.

Листинг 5. Код программы задания 5.

using System;

using System.Collections.Generic;

namespace Задание\_1.\_2.\_5

{

class Program

{

static void Main()

{

Console.WriteLine("Программа для работы с символьными массивами");

Console.WriteLine("Создание массива случайных букв и выделение согласных");

// Строки с буквами русского алфавита (строчные и заглавные)

string russianAlphabetLower = "абвгдеёжзийклмнопрстуфхцчшщъыьэюя";

string russianAlphabetUpper = "АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ";

string russianAlphabet = russianAlphabetLower + russianAlphabetUpper;

// Согласные буквы (строчные и заглавные)

string consonantsLower = "бвгджзйклмнпрстфхцчшщ";

string consonantsUpper = "БВГДЖЗЙКЛМНПРСТФХЦЧШЩ";

string consonants = consonantsLower + consonantsUpper;

while (true)

{

Console.Write("\nВведите количество элементов K (или 'q' для выхода): ");

string input = Console.ReadLine();

if (input.ToLower() == "q")

{

Console.WriteLine("Программа завершена. До свидания!");

break;

}

if (int.TryParse(input, out int k) && k > 0)

{

// Создаем исходный массив

char[] originalArray = CreateRandomCharArray(k, russianAlphabet);

// Создаем массив согласных

char[] consonantsArray = ExtractConsonants(originalArray, consonants);

// Выводим результаты

Console.WriteLine("\nИсходный массив:");

PrintCharArray(originalArray);

Console.WriteLine("\nМассив согласных букв:");

PrintCharArray(consonantsArray);

// Дополнительная информация

Console.WriteLine($"\nСтатистика:");

Console.WriteLine($"Всего элементов: {originalArray.Length}");

Console.WriteLine($"Согласных букв: {consonantsArray.Length}");

Console.WriteLine($"Гласных букв: {CountVowels(originalArray, consonants)}");

Console.WriteLine($"Других символов: {originalArray.Length - consonantsArray.Length - CountVowels(originalArray, consonants)}");

}

else

{

Console.WriteLine("Ошибка: K должно быть положительным целым числом.");

}

}

Console.WriteLine("\nНажмите любую клавишу для выхода...");

Console.ReadKey();

}

// Метод для создания массива случайных букв русского алфавита

static char[] CreateRandomCharArray(int k, string alphabet)

{

Random random = new Random();

char[] array = new char[k];

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

{

int randomIndex = random.Next(0, alphabet.Length);

array[i] = alphabet[randomIndex];

}

return array;

}

// Метод для извлечения согласных букв из массива

static char[] ExtractConsonants(char[] array, string consonants)

{

List<char> consonantsList = new List<char>();

foreach (char c in array)

{

if (consonants.Contains(c.ToString()))

{

consonantsList.Add(c);

}

}

return consonantsList.ToArray();

}

// Метод для подсчета гласных букв

static int CountVowels(char[] array, string consonants)

{

int count = 0;

string russianLetters = "абвгдеёжзийклмнопрстуфхцчшщъыьэюяАБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ";

foreach (char c in array)

{

if (russianLetters.Contains(c.ToString()) && !consonants.Contains(c.ToString()))

{

count++;

}

}

return count;

}

// Метод для вывода символьного массива

static void PrintCharArray(char[] array)

{

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

{

Console.Write($"{array[i]} ");

if ((i + 1) % 15 == 0)

Console.WriteLine();

}

if (array.Length % 15 != 0)

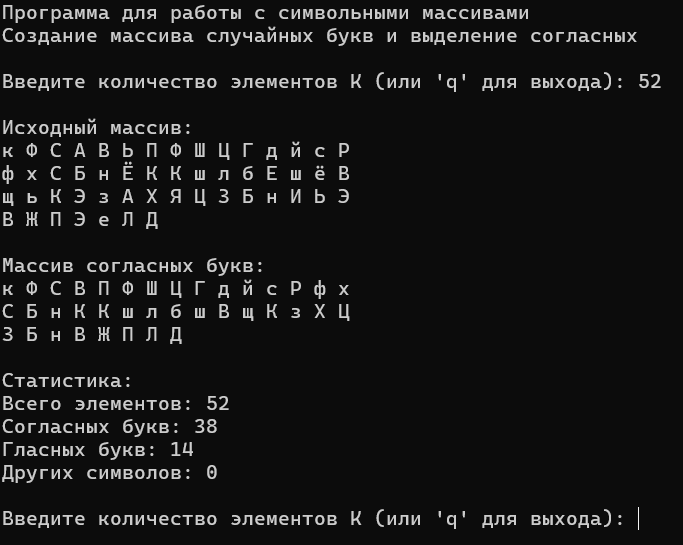
Console.WriteLine();

}

}

}

На рисунке 5 представлен результат работы программы.



**Задача 6.**

Определить вещественный массив из 10-ти элементов. Присвоить элементам случайные значения из диапазона [–10, 10). Сформировать массив индексов, которые нумеруют элементы первого массива в порядке возрастания их значений.

В листинге 6 представлен код программы.

Листинг 6. Код программы задания 6.

using System;

namespace Задание\_1.\_2.\_6

{

class Program

{

static void Main()

{

Console.WriteLine("Программа для работы с вещественным массивом");

Console.WriteLine("Создание массива индексов, упорядоченных по возрастанию значений элементов");

// Создаем и заполняем массив из 10 элементов

double[] array = CreateRandomArray(10, -10, 10);

// Выводим исходный массив

Console.WriteLine("\nИсходный массив:");

PrintArray(array);

// Создаем массив индексов, упорядоченных по возрастанию значений элементов

int[] sortedIndices = CreateSortedIndices(array);

// Выводим массив индексов

Console.WriteLine("\nМассив индексов, упорядоченных по возрастанию значений:");

PrintIndices(sortedIndices);

// Выводим значения элементов в порядке возрастания (для проверки)

Console.WriteLine("\nЭлементы массива в порядке возрастания:");

PrintSortedValues(array, sortedIndices);

// Дополнительная информация

Console.WriteLine($"\nМинимальный элемент: array[{sortedIndices[0]}] = {array[sortedIndices[0]]:F4}");

Console.WriteLine($"Максимальный элемент: array[{sortedIndices[sortedIndices.Length - 1]}] = {array[sortedIndices[sortedIndices.Length - 1]]:F4}");

Console.WriteLine("\nНажмите любую клавишу для выхода...");

Console.ReadKey();

}

// Метод для создания массива со случайными вещественными числами в диапазоне [min, max)

static double[] CreateRandomArray(int size, double min, double max)

{

Random random = new Random();

double[] array = new double[size];

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

{

// Генерируем случайное вещественное число в заданном диапазоне

array[i] = min + (random.NextDouble() \* (max - min));

}

return array;

}

// Метод для создания массива индексов, упорядоченных по возрастанию значений элементов

static int[] CreateSortedIndices(double[] array)

{

// Создаем массив индексов [0, 1, 2, ..., n-1]

int[] indices = new int[array.Length];

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

{

indices[i] = i;

}

// Сортируем индексы по значениям элементов массива

Array.Sort(indices, (a, b) => array[a].CompareTo(array[b]));

return indices;

}

// Метод для вывода массива индексов

static void PrintIndices(int[] indices)

{

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

{

Console.Write($"{indices[i],3}");

if ((i + 1) % 10 == 0)

Console.WriteLine();

}

if (indices.Length % 10 != 0)

Console.WriteLine();

}

// Метод для вывода значений массива в порядке возрастания (для проверки)

static void PrintSortedValues(double[] array, int[] sortedIndices)

{

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

{

int index = sortedIndices[i];

Console.Write($"{array[index]:F4} ");

if ((i + 1) % 5 == 0)

Console.WriteLine();

}

if (sortedIndices.Length % 5 != 0)

Console.WriteLine();

}

// Метод для вывода вещественного массива

static void PrintArray(double[] array)

{

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

{

Console.Write($"{array[i],8:F4}");

if ((i + 1) % 5 == 0)

Console.WriteLine();

}

if (array.Length % 5 != 0)

Console.WriteLine();

}

}

}

На рисунке 6 представлен результат работы программы.

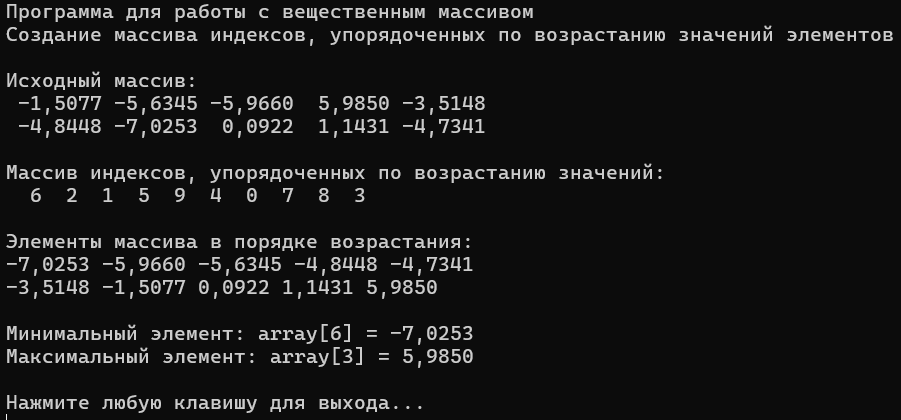


Рисунок 6 ­ Результат работы программы

**Вывод:** разработан модуль 1.2.