**НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ** **УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ** **«МОСКОВСКИЙ ФИНАНСОВО-ПРОМЫШЛЕННЫЙ УНИВЕРСИТЕТ** **“СИНЕРГИЯ”»**

|  |  |  |
| --- | --- | --- |
| **Факультет/Институт** |  | Информационных технологий |
|  |  | (наименование факультета/ Института) |
| **Направление/специальность** |  | Информационные системы и технологии |
| **подготовки:** |  | (код и наименование направления /специальности подготовки) |
| **Форма обучения:** |  | Очная |
|  |  | (очная, очно-заочная, заочная) |
|  |  |  |

**Отчет по практической работе №4**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **на тему** |  | Массивы и строки | | |
|  |  | (наименование темы) | | |
|  |  |  | | |
| **по дисциплине** | | |  | Разработка программных модулей |
|  | | |  | (наименование дисциплины) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Обучающийся** |  | Федотов П.Н. |  | Не требуется |
|  |  | (ФИО) |  | (подпись) |
| **Группа** |  | ДКИП-112 прог |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Преподаватель** |  | Сибирев И.В. |  |  |
|  |  | (ФИО) |  | (подпись) |

**Москва 2024 г.**

385 - Лабораторная работа 5. Одномерные массивы.  
389 - Лабораторная работа 6. Двумерные массивы  
393 - Лабораторная работа 7. Строки

**Вариант 13**

**Лабораторная 5 стр385**

**В одномерном массиве, состоящем из п вещественных элементов, вычислить:**

* **количество элементов массива, лежащих в диапазоне от А до В;**
* **сумму элементов массива, расположенных после максимального элемента.**

**Упорядочить элементы массива по убыванию модулей.**

**Листинг кода:**

using System;

class Program

{

static void Main()

{

Console.Write("Введите количество элементов в массиве: ");

int n = int.Parse(Console.ReadLine());

double[] array = new double[n];

Console.WriteLine("Введите элементы массива:");

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

{

array[i] = double.Parse(Console.ReadLine());

}

Console.Write("Введите значение A: ");

double A = double.Parse(Console.ReadLine());

Console.Write("Введите значение B: ");

double B = double.Parse(Console.ReadLine());

int countInRange = CountElementsInRange(array, A, B);

Console.WriteLine($"Количество элементов в диапазоне [{A}, {B}]: {countInRange}");

double sumAfterMax = SumAfterMax(array);

Console.WriteLine($"Сумма элементов после максимального элемента: {sumAfterMax}");

Array.Sort(array, (x, y) => Math.Abs(y).CompareTo(Math.Abs(x)));

Console.WriteLine("Отсортированный массив по убыванию модулей:");

foreach (var item in array)

{

Console.WriteLine(item);

}

}

static int CountElementsInRange(double[] array, double A, double B)

{

int count = 0;

foreach (var item in array)

{

if (item >= A && item <= B)

{

count++;

}

}

return count;

}

static double SumAfterMax(double[] array)

{

double sum = 0;

int maxIndex = Array.IndexOf(array, Max(array));

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

{

sum += array[i];

}

return sum;

}

static double Max(double[] array)

{

double max = array[0];

foreach (var item in array)

{

if (item > max)

{

max = item;

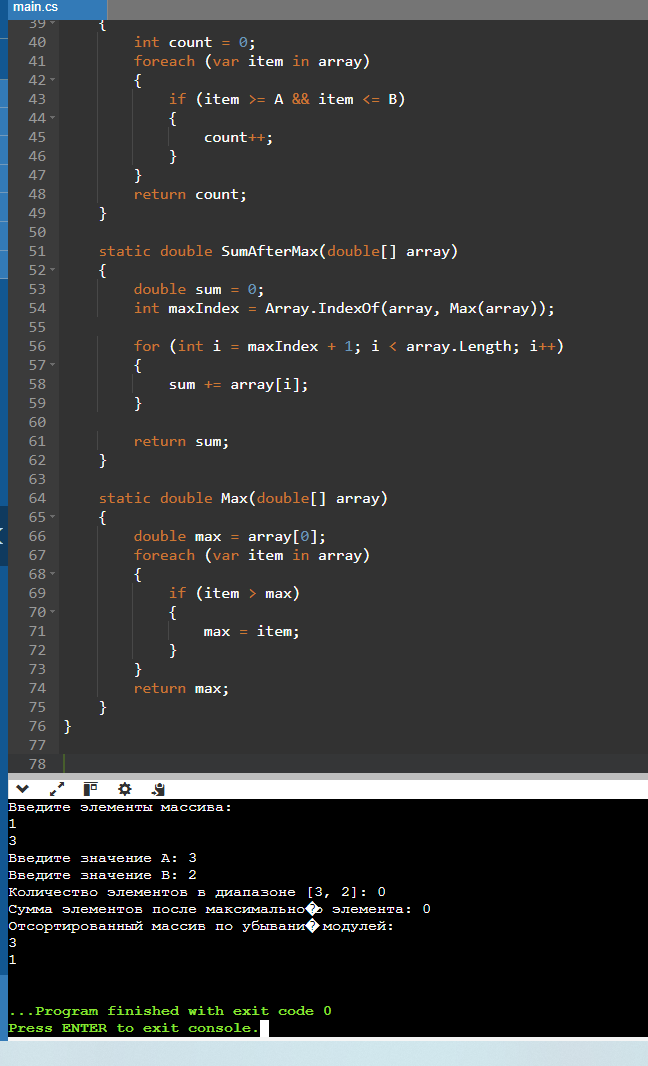
}

}

return max;

}

}



**389 - Лабораторная работа 6. Двумерные массивы  
вариант 13  
Осуществить циклический сдвиг элементов прямоугольной матрицы на п элементов вправо или вниз (в зависимости от введенного режима). может быть больше количества элементов в строке или столбце.**

**Листинг кода:**

using System;

class Program

{

static void Main()

{

Console.Write("Введите количество строк матрицы: ");

int rows = int.Parse(Console.ReadLine());

Console.Write("Введите количество столбцов матрицы: ");

int cols = int.Parse(Console.ReadLine());

int[,] matrix = new int[rows, cols];

Console.WriteLine("Введите элементы матрицы:");

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

{

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

{

matrix[i, j] = int.Parse(Console.ReadLine());

}

}

Console.Write("Введите количество сдвигов: ");

int shifts = int.Parse(Console.ReadLine());

Console.Write("Введите режим сдвига (1 - вправо, 2 - вниз): ");

int mode = int.Parse(Console.ReadLine());

if (mode == 1)

{

ShiftRight(matrix, shifts);

}

else if (mode == 2)

{

ShiftDown(matrix, shifts);

}

else

{

Console.WriteLine("Неверный режим сдвига!");

return;

}

Console.WriteLine("Результат после сдвига:");

PrintMatrix(matrix);

}

static void ShiftRight(int[,] matrix, int shifts)

{

int rows = matrix.GetLength(0);

int cols = matrix.GetLength(1);

// Уменьшаем количество сдвигов до размера столбца

shifts = shifts % cols;

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

{

int[] tempRow = new int[cols];

// Копируем текущую строку в временный массив

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

{

tempRow[j] = matrix[i, j];

}

// Сдвигаем элементы вправо

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

{

matrix[i, (j + shifts) % cols] = tempRow[j];

}

}

}

static void ShiftDown(int[,] matrix, int shifts)

{

int rows = matrix.GetLength(0);

int cols = matrix.GetLength(1);

shifts = shifts % rows;

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

{

int[] tempCol = new int[rows];

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

{

tempCol[i] = matrix[i, j];

}

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

{

matrix[(i + shifts) % rows, j] = tempCol[i];

}

}

}

static void PrintMatrix(int[,] matrix)

{

int rows = matrix.GetLength(0);

int cols = matrix.GetLength(1);

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

{

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

{

Console.Write(matrix[i, j] + "\t");

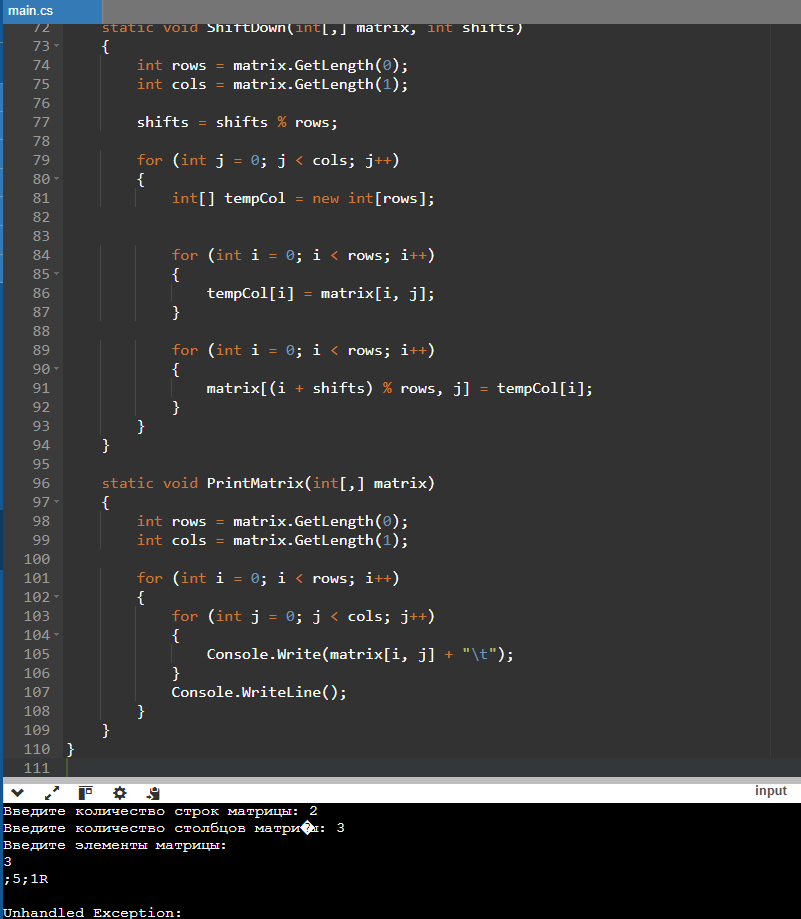
}

Console.WriteLine();

}

}

}



**393 - Лабораторная работа 7. Строки**

**Вариант 13**

**Написать программу, которая считывает английский текст из файла и выводит его на экран, заменив прописной каждую первую букву слов, начинающихся с гласной буквы.**

**Листинг кода:**

using System;

using System.IO;

class Program

{

static void Main()

{

string filePath = "path/to/your/file.txt";

try

{

string text = File.ReadAllText(filePath);

string modifiedText = CapitalizeFirstLetterOfVowelWords(text);

Console.WriteLine(modifiedText);

}

catch (Exception ex)

{

Console.WriteLine($"Произошла ошибка: {ex.Message}");

}

}

static string CapitalizeFirstLetterOfVowelWords(string input)

{

string[] words = input.Split(new char[] { ' ', '\n', '\r', '.', ',', '!', '?' }, StringSplitOptions.RemoveEmptyEntries);

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

{

if (IsVowel(words[i][0]))

{

words[i] = char.ToUpper(words[i][0]) + words[i].Substring(1);

}

}

return string.Join(" ", words);

}

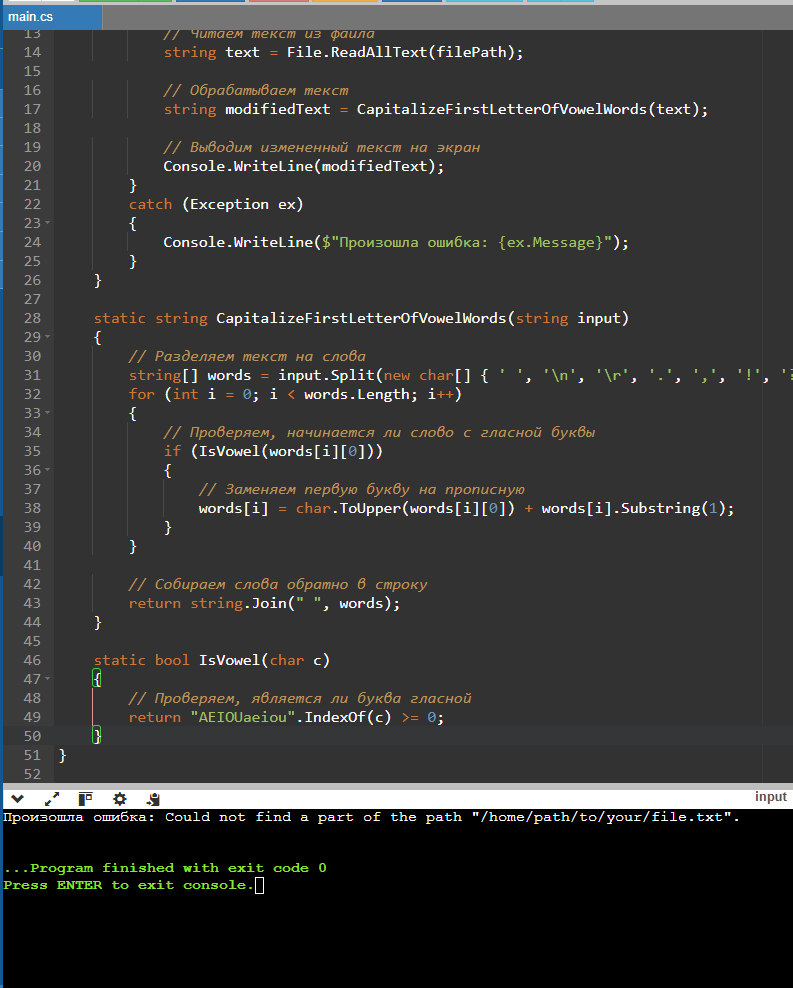
static bool IsVowel(char c)

{

return "AEIOUaeiou".IndexOf(c) >= 0;

}

}

****