Университет Синергия

Лабораторная работа № 4

Тема: Разработка программных модулей

Петров Антон ДКИП-203

Москва 2024

**Вариант 8**

**ЛР5 Задание 1:**

В одномерном массиве, состоящем из п вещественных элементов, вычислить: номер минимального элемента массива; сумму элементов массива, расположенных между первым и вторым отрицательными элементами. Преобразовать массив таким образом, чтобы сначала располагались все элементы, модуль которых не превышает единицу, а потом - все остальные

**Код:**

****

using System;

class Program

{

static void Main()

{

Console.Write("Введите количество элементов массива: ");

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

if (n <= 0)

{

Console.WriteLine("Размер массива должен быть положительным.");

return;

}

double[] arr = new double[n];

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

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

{

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

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

}

int minIndex = 0;

for (int i = 1; i < n; i++)

{

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

minIndex = i;

}

Console.WriteLine($"\nНомер минимального элемента: {minIndex} (значение {arr[minIndex]})");

int firstNegIndex = -1;

int secondNegIndex = -1;

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

{

if (arr[i] < 0)

{

if (firstNegIndex == -1)

firstNegIndex = i;

else

{

secondNegIndex = i;

break;

}

}

}

double sumBetweenNeg = 0;

if (firstNegIndex != -1 && secondNegIndex != -1 && secondNegIndex - firstNegIndex > 1)

{

for (int i = firstNegIndex + 1; i < secondNegIndex; i++)

{

sumBetweenNeg += arr[i];

}

Console.WriteLine($"Сумма элементов между первым и вторым отрицательными элементами: {sumBetweenNeg}");

}

else

{

Console.WriteLine("В массиве нет двух отрицательных элементов или они идут подряд, сумма равна 0.");

}

double[] transformed = new double[n];

int idx = 0;

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

{

if (Math.Abs(arr[i]) <= 1)

{

transformed[idx++] = arr[i];

}

}

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

{

if (Math.Abs(arr[i]) > 1)

{

transformed[idx++] = arr[i];

}

}

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

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

{

Console.Write(transformed[i] + " ");

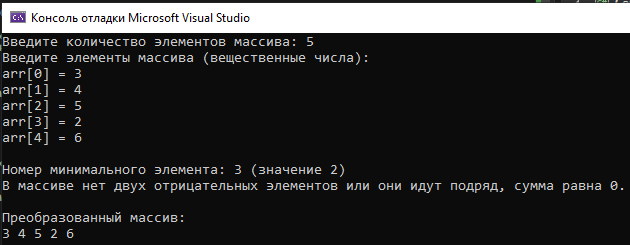
}

Console.WriteLine();

}

}

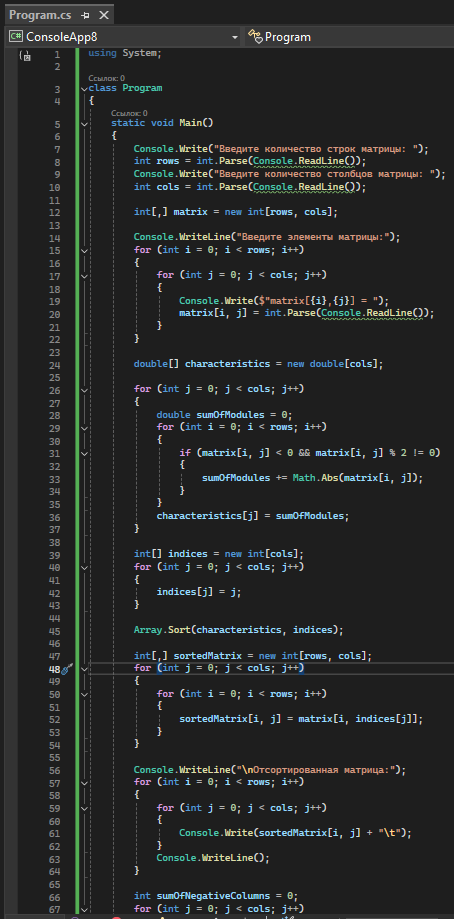
**Вывод:**

****

**ЛР6 Задание 2:**

Характеристикой столбца целочисленной матрицы назовем сумму модулей его отрицательных нечетных элементов, Переставляя столбцы заданной матрицы, расположить их в соответствии с ростом характеристик. Найти сумму элементов в тех столбцах, которые содержат хотя бы один отрицательный элемент.

**Код:**

****

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++)

{

Console.Write($"matrix[{i},{j}] = ");

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

}

}

double[] characteristics = new double[cols];

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

{

double sumOfModules = 0;

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

{

if (matrix[i, j] < 0 && matrix[i, j] % 2 != 0)

{

sumOfModules += Math.Abs(matrix[i, j]);

}

}

characteristics[j] = sumOfModules;

}

int[] indices = new int[cols];

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

{

indices[j] = j;

}

Array.Sort(characteristics, indices);

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

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

{

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

{

sortedMatrix[i, j] = matrix[i, indices[j]];

}

}

Console.WriteLine("\nОтсортированная матрица:");

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

{

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

{

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

}

Console.WriteLine();

}

int sumOfNegativeColumns = 0;

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

{

bool hasNegative = false;

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

{

if (sortedMatrix[i, j] < 0)

{

hasNegative = true;

break;

}

}

if (hasNegative)

{

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

{

sumOfNegativeColumns += sortedMatrix[i, j];

}

}

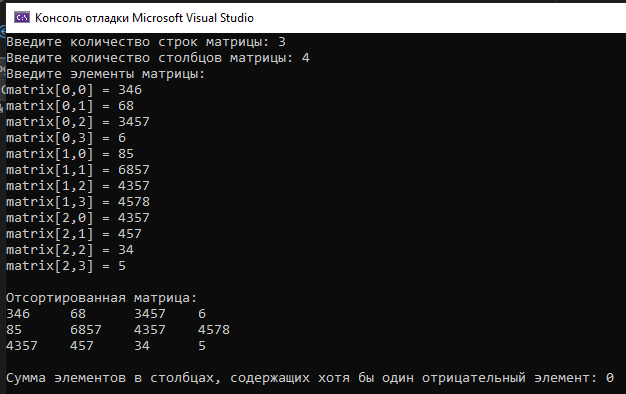
}

Console.WriteLine($"\nСумма элементов в столбцах, содержащих хотя бы один отрицательный элемент: {sumOfNegativeColumns}");

}

}

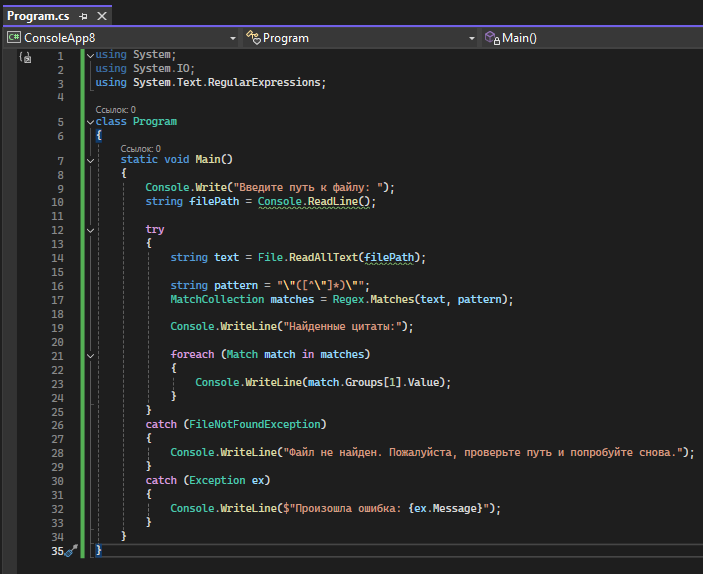
**Вывод:**

****

**ЛР7 Задание 3:**

Написать программу, которая считывает текст из файла и выводит на экран только цитаты, то есть предложения, заключенные в кавычки.

**Код:**

****

using System;

using System.IO;

using System.Text.RegularExpressions;

class Program

{

static void Main()

{

Console.Write("Введите путь к файлу: ");

string filePath = Console.ReadLine();

try

{

string text = File.ReadAllText(filePath);

string pattern = "\"([^\"]\*)\"";

MatchCollection matches = Regex.Matches(text, pattern);

Console.WriteLine("Найденные цитаты:");

foreach (Match match in matches)

{

Console.WriteLine(match.Groups[1].Value);

}

}

catch (FileNotFoundException)

{

Console.WriteLine("Файл не найден. Пожалуйста, проверьте путь и попробуйте снова.");

}

catch (Exception ex)

{

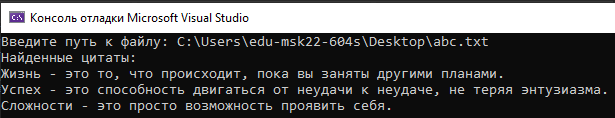
Console.WriteLine($"Произошла ошибка: {ex.Message}");

}

}

}

**Вывод:**

****