|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 **Прикладная информатика**

**Отчет**

**по лабораторной работе № 2**

**Дисциплина:** Разработка приложений на языке C#

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** \_\_\_\_\_\_\_\_\_\_\_\_\_

(Подпись, дата) (И.О. Фамилия)

Студент гр. ИУ6-72Б **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** И.С. Марчук

(Подпись, дата) (И.О. Фамилия)

Москва, 2022

**Цель работы:** в этой лабораторной работе обучающиеся закрепляют знания по C#, создавая консольное приложение. Программа представляет собой автоматизированную систему учета банковских сведений.

***Задание 1***: даны два массива a и b размерностью n и m соответственно, сформировать массив c таким образом, что первая часть — отсортированный по возрастанию массив а, а вторая часть — отсортированный по убыванию массив b.

***Задание 2***: создать двумерный массив, размерность задается пользователем, заполнить его случайными числами в диапазоне от 0 до 9. Отсортировать элементы массива по возрастанию вначале по строкам, а затем по столбцам. Вывести на экран исходный массив, массив отсортированный построчно, массив отсортированный по столбцам.

Код программы:

using System;

using System.Collections;

using System.Linq;

namespace LabApp2

{

internal class Program

{

static void Main(string[] args)

{

// Задание 1: Задание 1: Задание 1: Задание 1: Задание 1: Задание 1: Задание 1: Задание 1: Задание 1:

String buf;

Random rnd = new Random();

int[] a;

int[] b;

Console.Write("Задание 1! \nВведите размер a(число n):");

int n = 0;

buf = Console.ReadLine();

if (!int.TryParse(buf, out n) || n < 0)

{

Console.WriteLine("Strannoe chislo, vozimu десять");

n = 10;

}

Console.Write("Введите размер b(число m):");

int m = 0;

buf = Console.ReadLine();

if (!int.TryParse(buf, out m) || m < 0)

{

Console.WriteLine("Strannoe chislo, vozimu пять");

m = 10;

}

a = new int[n];

b = new int[m];

//..... generate random degenerate

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

{

a[aPoszss] = rnd.Next(-9, 99);

}

for (int bPoszss = 0; bPoszss < m; bPoszss++)

{

b[bPoszss] = rnd.Next(-9, 99);

}

// вывод

Console.Write("\na:");

arrrrrrrAyOuUUuuuUUUuuUUUuTTTtttTTTT(a);

Console.Write("b:");

arrrrrrrAyOuUUuuuUUUuuUUUuTTTtttTTTT(b);

// сортировочка

Array.Sort(a, new CompotTop());

Array.Sort(b, new CompotDown());

int[] аы = a.Concat(b).ToArray();

// вывод

Console.WriteLine();

arrrrrrrAyOuUUuuuUUUuuUUUuTTTtttTTTT(аы);

Console.WriteLine("ъуъ!");

Console.ReadLine();

// Задание 2: Задание 2: Задание 2: Задание 2: Задание 2: Задание 2: Задание 2: Задание 2: Задание 2:

Console.Write("Задание 2! \nВведите размер массива по горизонтали:");

Console.ReadLine();

Console.Write("А точно вы же уже ввели два числа. Вот:\n\nМассив "+ n +" на "+m+":\n");

int[][] matrix = new int[m][];

// generate random Y'arr

for (int ugrek = 0; ugrek < m; ugrek++)

{

matrix[ugrek] = new int[n];

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

{

matrix[ugrek][ex] = rnd.Next(0, 9);

}

}

// вывод

outArrDoubleDimensoinOut(matrix);

Console.WriteLine("\nСортируем по строкам:");

for (int ugrek = 0; ugrek < m; ugrek++)

{

matrix[ugrek] = sortArray(matrix[ugrek], 0, matrix[ugrek].Length-1);

}

// вывод

outArrDoubleDimensoinOut(matrix);

Console.WriteLine("\nА теперь по столбцам:");

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

matrix = sortArray(matrix, columnI, 0, m - 1);

// вывод

outArrDoubleDimensoinOut(matrix);

Console.ReadLine();

}

// вывод массива

static void arrrrrrrAyOuUUuuuUUUuuUUUuTTTtttTTTT(int [] аы)

{

Console.Write("[");

for (int arrayPos = 0; arrayPos < аы.Length; arrayPos++)

{

Console.Write(аы[arrayPos] + (arrayPos == аы.Length - 1 ? "" : ", "));

}

Console.WriteLine("]");

}

static void outArrDoubleDimensoinOut(int[][] arr) {

for (int arrayPos = 0; arrayPos < arr.Length; arrayPos++)

{

Console.Write(arrayPos+".");

arrrrrrrAyOuUUuuuUUUuuUUUuTTTtttTTTT(arr[arrayPos]);

}

}

// быстрая сортировка

public static int[] sortArray(int[] array, int leftIndex, int rightIndex)

{

var i = leftIndex;

var j = rightIndex;

var pivot = array[leftIndex];

while (i <= j)

{

while (array[i] < pivot)

{

i++;

}

while (array[j] > pivot)

{

j--;

}

if (i <= j)

{

int temp = array[i];

array[i] = array[j];

array[j] = temp;

i++;

j--;

}

}

if (leftIndex < j)

sortArray(array, leftIndex, j);

if (i < rightIndex)

sortArray(array, i, rightIndex);

return array;

}

// быстрая сортировка для матрицы по столбцам

public static int[][] sortArray(int[][] arr, int column, int leftIndex, int rightIndex) {

var i = leftIndex;

var j = rightIndex;

var pivot = arr[leftIndex][column];

while (i <= j)

{

while (arr[i][column] < pivot)

i++;

while (arr[j][column] > pivot)

j--;

if (i <= j)

{

int temp = arr[i][column];

arr[i][column] = arr[j][column];

arr[j][column] = temp;

i++;

j--;

}

}

if (leftIndex < j)

sortArray(arr, column, leftIndex, j);

if (i < rightIndex)

sortArray(arr, column, i, rightIndex);

return arr;

}

}

// компараторы

public class CompotTop : IComparer

{

public int Compare(object x, object y)

{

return ((int)x - (int)y);

}

}

public class CompotDown : IComparer

{

public int Compare(object x, object y)

{

return ((int)y - (int)x);

}

}

}

Работа программы показана на рисунках 1 - 2.

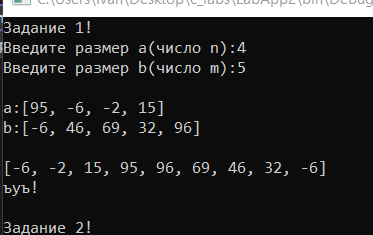


Рисунок 1 – работа задания 1

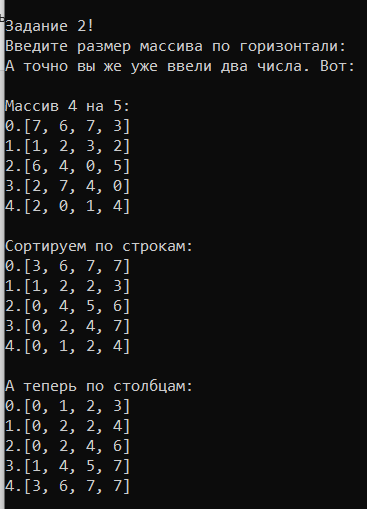


Рисунок 2 – Работа задания 2

**Вывод:** в ходе лабораторной работы я научился работать с одномерными массивами и матрицами на языке C#. Сгенерированные случайным образом массивы были отсортированы по горизонтали и вертикали (для двумерных массивов). По возрастанию и убыванию. Также был освоен вывод команд на экран.