



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 09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

О Т Ч Е Т

по лабораторной работе № 2

Название: Массивы

Дисциплина: Разработка приложений на языке C#

Студент

ИУ6-75Б

(Группа)

(Подпись, дата)

И.Ю. Жосан

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

А.М. Минитаева

(И.О. Фамилия)

Москва, 2023

Цель работы: изучить основы работы с массивами в языке C# на практических задачах.

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 *a*, а вторая часть — отсортированный по убыванию массив *b*.

Выполнение задания 1:

Алгоритм предусматривает ввод с клавиатуры размеров массивов *a* и *b* (положительное целое число), а также содержание этих массивов (целые числа). После этого производится сортировка введенных массивов и их конкатенация.

Код разработанной программы:

```
using System;

namespace lab2_array_task1
{
    class Program
    {
        static void Main(string[] args)
        {
            int n, m;
            Console.WriteLine("Enter size of array A:");
            while (true)
            {
                while (!(int.TryParse(Console.ReadLine(), out n)))
                    Console.WriteLine("ERROR: Wrong number. Enter size of array A again:");
                if (n < 0)
                    Console.WriteLine("ERROR: Number less than zero. Enter size of array A again:");
                else
                    break;
            }
            int[] a = new int[n];
            for (int i = 0; i < n; i++)
            {
                Console.WriteLine($"Enter A[{i}]:");
                while (!(int.TryParse(Console.ReadLine(), out a[i])))
                    Console.WriteLine($"ERROR: Wrong number. Enter A[{i}] again:");
            }

            Console.WriteLine("Enter size of array B:");
            while (true)
            {
                while (!(int.TryParse(Console.ReadLine(), out m)))
                    Console.WriteLine("ERROR: Wrong number. Enter size of array B again:");
            }
        }
    }
}
```

```

        if (m < 0)
            Console.WriteLine("ERROR: Number less then zero.
                                Enter size of array B again:");
        else
            break;
    }
    int[] b = new int[m];
    for (int i = 0; i < m; i++)
    {
        Console.WriteLine($"Enter B[{i}]:");
        while (!(int.TryParse(Console.ReadLine(), out b[i])))
            Console.WriteLine($"ERROR: Wrong number. Enter
                                B[{i}] again:");
    }

    Console.WriteLine("\nArray A: " + String.Join(", ", a));
    Console.WriteLine("Array B: " + String.Join(", ", b));

    Array.Sort(a);
    Array.Sort(b);
    Array.Reverse(b);

    Console.WriteLine("Sorted array A: " + String.Join(", ",
        a));
    Console.WriteLine("Sorted array B: " + String.Join(", ",
        b));

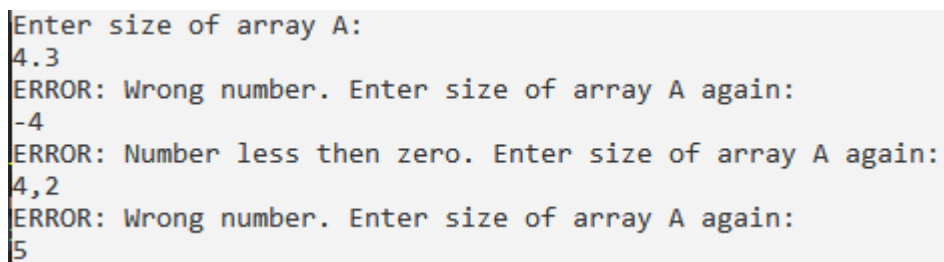
    int[] c = new int[n + m];
    a.CopyTo(c, 0);
    b.CopyTo(c, a.Length);

    Console.WriteLine("RESULT - Array C: " + String.Join(", ",
        c));
    Console.ReadLine();
}
}
}

```

Рисунки, отображающие работу программы, приведены ниже:

- рисунок 1 – защита от неправильного ввода числа - размера массива;
- рисунок 2 – защита от неправильного ввода числа - элемента массива;
- рисунок 3 – пример сортировки массивов и их конкатенации.



```

Enter size of array A:
4.3
ERROR: Wrong number. Enter size of array A again:
-4
ERROR: Number less then zero. Enter size of array A again:
4,2
ERROR: Wrong number. Enter size of array A again:
5

```

Рисунок 1 – Защита от неправильного ввода размера массива

```

Enter A[0]:
r
ERROR: Wrong number. Enter A[0] again:
-3
Enter A[1]:
6
Enter A[2]:
1
Enter A[3]:
-87
Enter A[4]:
8

```

Рисунок 2 – Защита от неправильного ввода элементов массива

```

Array A: -3, 6, 1, -87, 8
Array B: 44, 66
Sorted array A: -87, -3, 1, 6, 8
Sorted array B: 66, 44
RESULT - Array C: -87, -3, 1, 6, 8, 66, 44

```

Рисунок 3 – Пример сортировки массивов и их конкатенации

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.

Выполнение задания 2:

Алгоритм предусматривает ввод с клавиатуры размера двумерного массива n и m (положительные целые числа). Далее массив (массив размером n массивов размером m) наполняется случайными целыми числами в диапазоне от 0 до 9.

Для удобной сортировки по столбцам создается транспонированная матрица (массив размером m массивов размером n). После чего производится сортировка обоих двумерных массивов и их вывод в терминал.

Код разработанной программы:

```

using System;
namespace lab2_array_task2
{
    class Program
    {
        static void Main(string[] args)
        {
            int n, m;
            Console.WriteLine("Enter number of ROWS n in matrix");
            Console.WriteLine("Enter number of COLUMNS m in matrix");
            while (true)
            {
                while (!(int.TryParse(Console.ReadLine(), out n)))
                {
                    Console.WriteLine("Invalid input. Please enter a valid number for rows.");
                }
                while (!(int.TryParse(Console.ReadLine(), out m)))
                {
                    Console.WriteLine("Invalid input. Please enter a valid number for columns.");
                }
                break;
            }
        }
    }
}

```

```

        Console.WriteLine("ERROR: Wrong number. Enter
            number of rows again:");
    if (n < 0)
        Console.WriteLine("ERROR: Number less than zero.
            Enter number of rows again:");
    else
        break;
}
Console.WriteLine("Enter number of COLUMNS m in matrix:");
while (true)
{
    while (!(int.TryParse(Console.ReadLine(), out m)))
        Console.WriteLine("ERROR: Wrong number. Enter
            number of columns again:");
    if (m < 0)
        Console.WriteLine("ERROR: Number less than zero.
            Enter number of columns again:");
    else
        break;
}
int[][] a = new int[n][], b = new int[m][];
Random rand = new Random();
Console.WriteLine("\nArray A: ");
for (int i = 0; i < n; i++)
{
    a[i] = new int[m];
    for (int j = 0; j < m; j++)
        a[i][j] = rand.Next(0, 9);
    Console.WriteLine(String.Join(" ", a[i]));
}
for (int i = 0; i < m; i++)
{
    b[i] = new int[n];
    for (int j = 0; j < n; j++)
        b[i][j] = a[j][i];
    Array.Sort(b[i]);
}
Console.WriteLine("\nSorted by rows array A: ");
for (int i = 0; i < n; i++)
{
    Array.Sort(a[i]);
    Console.WriteLine(String.Join(" ", a[i]));
}
Console.WriteLine("\nSorted by columns array A: ");
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < m; j++)
        a[i][j] = b[j][i];
    Console.WriteLine(String.Join(" ", a[i]));
}
Console.ReadLine();
}
}
}

```

Пример защиты от неправильного ввода числа - размера массива – приведен на рисунке 1.

```
Enter number of ROWS n in matrix A[n,m]:  
-  
ERROR: Wrong number. Enter number of rows again:  
ы  
ERROR: Wrong number. Enter number of rows again:  
-2  
ERROR: Number less then zero. Enter number of rows again:  
5  
Enter number of COLUMNS m in matrix A[n,m]:  
8
```

Рисунок 1 – Защита от неправильного ввода размера массива

Рисунок 2 приведен пример сортировки массива.

```
Array A:  
8 0 3 7 2 8 6 3  
4 7 6 6 8 7 6 2  
6 3 2 2 8 2 2 6  
0 2 5 7 2 1 5 4  
2 5 4 7 2 6 0 4  
  
Sorted by rows array A:  
0 2 3 3 6 7 8 8  
2 4 6 6 6 7 7 8  
2 2 2 2 3 6 6 8  
0 1 2 2 4 5 5 7  
0 2 2 4 4 5 6 7  
  
Sorted by columns array A:  
0 0 2 2 2 1 0 2  
2 2 3 6 2 2 2 3  
4 3 4 7 2 6 5 4  
6 5 5 7 8 7 6 4  
8 7 6 7 8 8 6 6
```

Рисунок 2 – Пример сортировки массивов и их конкатенации

Вывод: в процессе выполнения лабораторной работы были изучены основы массивов в языке программирование С# (создание n-мерных массивов, сортировка, обратное следование элементов, вывод в консоль).

В результате практической работы были получены две консольные программы, выполняющие проверки на правильность вводимых чисел и обрабатывающие массивы в соответствии с заданием.