

Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

ОТЧЕТ

ЛАБОРАТОРНАЯ РАБОТА №1

ССП

Выполнил
студент группы
АС-50:
Протасевич А.В.
Проверил:
Крощенко А.А.

Брест 2020

Цель работы: приобрести практические навыки обработки параметров командной строки, закрепить базовые знания языка программирования C# при решении практических задач.

Вариант 8

Задание 1

Для переданной в качестве параметра последовательности из N целых чисел написать утилиту с функционалом: Вывод всех уникальных чисел.

Аргументы командной строки:

0 1 2 3 3 5 7 7 9

Задание 2

Написать функцию, выполняющую указанную операцию над массивом. Использовать только базовые возможности языка, без привлечения специализированных функций для обработки коллекций. Ввод массивов выполнять из командной строки.

Напишите метод `randomPerturbations(double[][] matrix)`, который переставляет элементы матрицы `matrix` случайным образом.

Задание 3

Решите задачу на обработку строк. Ввод исходных строк выполнять из командной строки.

Напишите метод `boolean isAllLowerCase(String cs)`, проверяющий, все ли символы строки в нижнем регистре или нет.

`isAllLowerCase (null) = false`

`isAllLowerCase ("") = false`

`isAllLowerCase (" ") = false`

`isAllLowerCase (" abc ") = true`

`isAllLowerCase (" abC ") = false`

Код программы:

```
using System;
using System.Collections.Generic;

namespace lab1SSP
{
    class Program
    {
        static int x, y;
        static void Main(string[] args)
        {
            Console.WriteLine("Result of Task 1:\n");
            Task1(args);

            Console.WriteLine("Task 2:\n");
            Console.WriteLine("Enter count lines ");
            int.TryParse(Console.ReadLine(), out x);
            Console.WriteLine("Enter count columns ");
            int.TryParse(Console.ReadLine(), out y);
            double[,] array = new double[x, y];
```

```

Console.WriteLine($"Enter matrix {x}x{y}");
try
{
    int count = 0;
    string enterString = Console.ReadLine();
    string[] arrayString = enterString.Split(new char[] { ' ' });
    for (int i = 0; i < x; i++)
    {
        for (int j = 0; j < y; j++)
        {
            array[i, j] = Convert.ToDouble(arrayString[count]);
            count++;
        }
    }
}
catch (FormatException)
{
    Console.WriteLine("invalid format");
}
catch (IndexOutOfRangeException)
{
    Console.WriteLine("index out of range");
}
randomPerturbations(ref array);
Console.WriteLine();
for (int i = 0; i < x; i++)
{
    for (int j = 0; j < y; j++)
    {
        Console.Write($"{array[i, j]} ");
    }
    Console.WriteLine();
}
Console.WriteLine();
Console.Write("Result of Task 3:\n");
String a;
do
{
    Console.Write("Enter value\n");
    a = Console.ReadLine();
    Console.WriteLine($"{a} is Lower - {isAllLowerCase(a)}");
} while (a != "0");
Console.ReadKey();
}
static void Task1(string[] args)
{
    try
    {
        List<int> array = new List<int>(); ;
        foreach (var item in args)
        {
            if (int.Parse(item) is int)
                array.Add(Convert.ToInt32(item));
            else
                throw new FormatException("Not all args is int");
        }
        if (array == null)
            throw new ArgumentNullException("Args Nullable");
        for (int i = 0; i < array.Count - 1; i++)
        {
            for (int j = i + 1; j < array.Count; j++)
            {
                if (array[i] == array[j])
                {
                    array.RemoveAt(j);
                }
            }
        }
    }
    catch { }
}

```

```

        array.RemoveAt(i);
        continue;
    }
}
Console.WriteLine(array[i]);
}
}
catch (FormatException ex)
{
    Console.WriteLine(ex.Message);
}
catch (ArgumentNullException ex)
{
    Console.WriteLine(ex.Message);
}
}
static void randomPerturbations(ref double[, ] args)
{
    Random r = new Random();
    int count = r.Next(0, args.Length - 1);
    Console.WriteLine();
    Console.WriteLine($"Count random {count}");
    while (count != 0)
    {
        int x1 = r.Next(0, x);
        int y1 = r.Next(0, y);
        int x2 = r.Next(0, x);
        int y2 = r.Next(0, y);
        var buffer = args[x1, y1];
        args[x1, y1] = args[x2, y2];
        args[x2, y2] = buffer;
        count--;
    }
}
static bool isAllLowerCase(String cs)
{
    if (cs == null || cs == "" || cs == " ")
        return false;
    if (cs == cs.ToLower())
        return true;
    return false;
}
}
}
}

```

Result of Task 1:

0

1

2

5

9

Task 2:

Enter count lines 3

Enter count columns 3

Enter matrix 3x3

1 2 3 4 5 6 7 8 9

Count random 2

1 8 3

4 5 9

7 2 6

Result of Task 3:

Enter value

NULL

NULL is Lower - False

Enter value

is Lower - False

Enter value

is Lower - False

Enter value

abc

abc is Lower - True

Enter value

abC

abC is Lower - False