

Міністерство освіти України  
Національний технічний університет "ХПІ"  
кафедра "Інформатики та інтелектуальної власності"

**Звіт**  
**Лабораторна робота 1**  
з дисципліни "DOT NET"

Виконав: студент групи КН-921Б  
Бірюков Д. Є.  
Перевірив:  
Івашко А.В.

## ЗМІСТ

Завдання 1 .....	3
Завдання 2 .....	5
Завдання 3 .....	7
Завдання 4 .....	9
Завдання 5 .....	11
Завдання 6 .....	12
Завдання 7 .....	12
Завдання 8 .....	13
Завдання 9 .....	15
Завдання 10 .....	16
Висновок .....	18

## Завдання 1

Використовуючи завдання власного варіанта до лабораторної роботи 2 з електронних методичних вказівок з основ програмування та алгоритмічних мов створити проект з реалізації цих завдань засобами мови C#.

Разработать программу, которая вводит фактические данные из таблицы, представленной в Вашем варианте индивидуального задания и выводит на экран таблицу, подобную той, которая находится в индивидуальном задании (включая заголовок и примечания).

Фирмы - производители СКБД			
Фирма	Количество. продуктов	Годовой объем продаж (\$)	Часть рынка (%)
Oracle	1	2488000000	31.1
IBM	3	2392000000	29.9
Microsoft	2	1048000000	13.1
Примечание: по данным Gartner Group за 1999г			

using System;

class Program

```
{
    static void Main()
    {
        // Заголовок таблицы
        string tableTitle = "Фирмы - производители СКБД";
        Console.WriteLine(tableTitle);

        // Двумерный массив для хранения данных о фирмах
        string[,] firmData = {
            { "Фирма", "Количество. продуктов", "Годовой объем продаж", "Часть рынка (%)" },
            { "Oracle", "1", "2488000000", "31.1" },
            { "IBM", "3", "2392000000", "29.9" },
            { "Microsoft", "2", "1048000000", "13.1" }
        };

        // Определение ширины столбцов на основе данных
        int[] columnWidths = new int[firmData.GetLength(1)];
        for (int i = 0; i < firmData.GetLength(0); i++)
        {
            for (int j = 0; j < firmData.GetLength(1); j++)
            {
                if (firmData[i, j].Length > columnWidths[j])
```

```

        {
            columnWidths[j] = firmData[i, j].Length;
        }
    }
}

```

```

// Вывод данных о фирмах с выравниванием
Console.WriteLine();

```

```

for (int i = 0; i < firmData.GetLength(0); i++)
{
    for (int j = 0; j < firmData.GetLength(1); j++)
    {
        Console.Write(firmData[i, j].PadRight(columnWidths[j] + 2));
    }
    Console.WriteLine();
}

```

```

// Примечание

```

```

Console.WriteLine("Примечание: по данным Gartner Group за 1999г");

```

```

}
}

```

```

Консоль отладки Microsoft V x + -
Фирмы - производители СКБД
Фирма    Количество. продуктов  Годовой объем продаж ($)  Часть рынка (%)
Oracle   1                      2488000000                31.1
IBM      3                      2392000000                29.9
Microsoft 2                      1048000000                13.1
Примечание: по данным Gartner Group за 1999г

C:\Users\dmitybirukov\source\repos\Lab_01\Task_01\bin\Debug\net6.0\Task_01.exe (процесс 5492) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

```

- Завдання 2\*

Виконати минуле завдання, покращивши зовнішній вигляд таблиці із вихідними результатами за допомогою символів псевдографіки (┌, ┐, └, ┘, ┌, ┐, └, ┘ та ін.) та налаштування кольорів консолі, наприклад, числа відобразити одним кольором, текст – іншим, таблицю – іншим від попередніх.

```
using System;
```

```
class Program
```

```
{
```

```
static void Main()
```

```
{
```

```
    // Заголовок таблицы
```

```
    string tableTitle = "Фирмы - производители СКБД";
```

```
    Console.ForegroundColor = ConsoleColor.Blue;
```

```
    Console.WriteLine(tableTitle);
```

```
    Console.ResetColor();
```

```
    // Двумерный массив для хранения данных о фирмах
```

```
    string[,] firmData = {
```

```
        { "Фирма", "Количество. продуктов", "Годовой объем продаж ($)", "Часть рынка (%)" },
```

```
        { "Oracle", "1", "2488000000", "31.1" },
```

```
        { "IBM", "3", "2392000000", "29.9" },
```

```
        { "Microsoft", "2", "1048000000", "13.1" }
    };
```

```
    // Определение ширины столбцов на основе данных
```

```
    int[] columnWidths = new int[firmData.GetLength(1)];
```

```
    for (int i = 0; i < firmData.GetLength(0); i++)
```

```
    {
```

```
        for (int j = 0; j < firmData.GetLength(1); j++)
```

```
        {
```

```
            if (firmData[i, j].Length > columnWidths[j])
```

```
            {
```

```
                columnWidths[j] = firmData[i, j].Length;
```

```
            }
```

```
        }
```

```
    }
```

```
    // Вывод данных о фирмах с псевдографикой и цветами
```

```

Console.WriteLine();
for (int i = 0; i < firmData.GetLength(0); i++)
{
    for (int j = 0; j < firmData.GetLength(1); j++)
    {
        Console.ForegroundColor = ConsoleColor.Yellow;
        Console.Write("┌" + new string('─', columnWidths[j] + 2) + "┐");
        Console.ResetColor();
    }
    Console.WriteLine();

    for (int j = 0; j < firmData.GetLength(1); j++)
    {
        Console.ForegroundColor = ConsoleColor.Yellow;
        Console.Write(" │ ");
        Console.ResetColor();
        Console.ForegroundColor = ConsoleColor.Blue;
        Console.Write(firmData[i, j].PadRight(columnWidths[j]));
        Console.ResetColor();
        Console.ForegroundColor = ConsoleColor.Yellow;
        Console.Write(" │ ");
        Console.ResetColor();
    }
    Console.WriteLine();

    for (int j = 0; j < firmData.GetLength(1); j++)
    {
        Console.ForegroundColor = ConsoleColor.Yellow;
        Console.Write("└" + new string('─', columnWidths[j] + 2) + "┘");
        Console.ResetColor();
    }
    Console.WriteLine();
}

// Примечание
Console.ForegroundColor = ConsoleColor.Blue;
Console.WriteLine("Примечание: по данным Gartner Group за 1999г");
Console.ResetColor();
}
}

```

Консоль отладки Microsoft V X + -

Фирмы – производители СКБД

Фирма	Количество, продуктов	Годовой объем продаж (\$)	Часть рынка (%)
Oracle	1	2488000000	31.1
IBM	3	2392000000	29.9
Microsoft	2	1048000000	13.1

Примечание: по данным Gartner Group за 1999г

### Завдання 3

Використовуючи завдання власного варіанта до лабораторної роботи 3 з електронних методичних вказівок з основ програмування та алгоритмічних мов створити проект з реалізації цих завдань засобами мови C#.

Составьте программу, которая подсчитывает и выводит значение  $t1$  и  $t2$  по формулам, которые приведены в Вашем варианте индивидуального задания. Определите области допустимых значений параметров формул и задайте произвольные значения из этих областей. Параметры, которые имеют имена:  $n$  и  $m$  - целые, остальные параметры - с плавающей точкой. Значения параметров с именами  $x$  и  $y$  должны вводиться с клавиатуры, значения остальных - задаваться как начальные значения при объявлении соответствующих переменных. Допускается (и даже желательно) упростить / разложить формулы для того, чтобы обеспечить минимизацию объема вычислений.

$$t1 = \frac{1}{c} \left[ \frac{b}{a} \ln(ax+b) + \frac{d}{y} \ln(yx+d) \right]$$

$$t2 = \frac{1}{a(n-1)} \frac{\sin ax}{\cos^{n-1} ax}$$

```
using System;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        // Ввод значений x и y с клавиатуры
```

```
        Console.Write("Введите значение x: ");
```

```
        double x = double.Parse(Console.ReadLine());
```

```
        Console.Write("Введите значение y: ");
```

```
        double y = double.Parse(Console.ReadLine());
```

```

// Задание начальных значений для параметров
double a = 2.0;
double b = 3.0;
double c = 1.0;
double d = 4.0;
int n = 3;
int m = 5;

// Вычисление t1 и t2
double t1 = CalculateT1(x, y, a, b, c, d);
double t2 = CalculateT2(x, a, c, n);

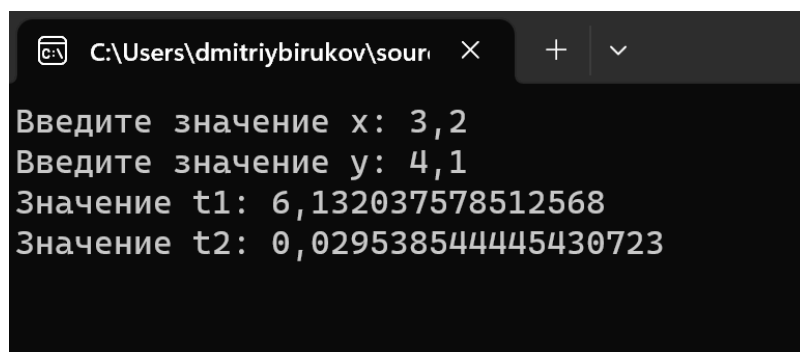
// Вывод результатов
Console.WriteLine($"Значение t1: {t1}");
Console.WriteLine($"Значение t2: {t2}");

Console.ReadLine();
}

// Функция для вычисления t1
static double CalculateT1(double x, double y, double a, double b, double c, double d)
{
    double term1 = (b / a) * Math.Log(a * x + b);
    double term2 = (d / y) * Math.Log(y * x + d);
    return 1.0 / c * Math.Abs(term1 + term2);
}

// Функция для вычисления t2
static double CalculateT2(double x, double a, double c, int n)
{
    return 1.0 / (a * (n - 1)) * Math.Sin(a * x) / Math.Pow(Math.Cos(a * x), n - 1);
}
}

```



```

C:\Users\dmitriybirukov\source\...
Введите значение x: 3,2
Введите значение y: 4,1
Значение t1: 6,132037578512568
Значение t2: 0,029538544445430723

```



## Завдання 4\*

Виконати минуле завдання, покращивши гнучкість програми обчислення виразу, якщо вхідні дані є некоректними із точки зору математичних правил (нуль в знаменнику, від'ємне значення під знаком квадратного кореня та ін.). Для цього використати механізм обробки відповідних виключень (try ... catch ... finally)

```
using System;

class Program
{
    static void Main()
    {
        try
        {
            // Ввод значений x и y с клавиатуры
            Console.Write("Введите значение x: ");
            double x = double.Parse(Console.ReadLine());

            Console.Write("Введите значение y: ");
            double y = double.Parse(Console.ReadLine());

            // Проверка на соответствие математическим правилам
            if (y == 0 || Math.Cos(x) == 0)
            {
                throw new ArgumentException("y не может быть нулем, и косинус(x) не может быть нулем.");
            }

            // Задание начальных значений для параметров
            double a = 2.0;
            double b = 3.0;
            double c = 1.0;
            double d = 4.0;
            int n = 3;
            int m = 5;

            // Вычисление t1 и t2
            double t1 = CalculateT1(x, y, a, b, c, d);
            double t2 = CalculateT2(x, a, c, n);

            // Вывод результатов
```

```

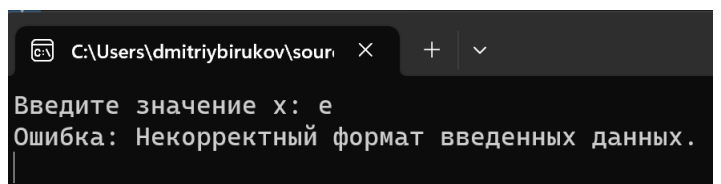
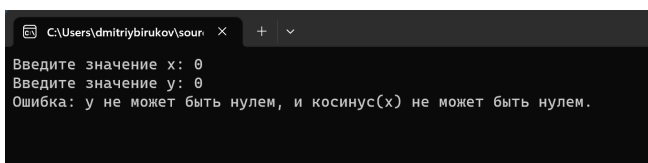
        Console.WriteLine($"Значение t1: {t1}");
        Console.WriteLine($"Значение t2: {t2}");
    }
    catch (FormatException)
    {
        Console.WriteLine("Ошибка: Некорректный формат введенных данных.");
    }
    catch (ArgumentException e)
    {
        Console.WriteLine($"Ошибка: {e.Message}");
    }
    catch (Exception e)
    {
        Console.WriteLine($"Ошибка: {e.Message}");
    }

    Console.ReadLine();
}

// Функция для вычисления t1
static double CalculateT1(double x, double y, double a, double b, double c, double d)
{
    double term1 = (b / a) * Math.Log(a * x + b);
    double term2 = (d / y) * Math.Log(y * x + d);
    return 1.0 / c * Math.Abs(term1 + term2);
}

// Функция для вычисления t2
static double CalculateT2(double x, double a, double c, int n)
{
    return 1.0 / (a * (n - 1)) * Math.Sin(a * x) / Math.Pow(Math.Cos(a * x), n - 1);
}
}

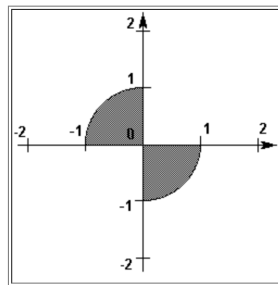
```



## Завдання 5

Використовуючи завдання власного варіанта до лабораторної роботи 5 з електронних методичних вказівок з основ програмування та алгоритмічних мов створити проект з реалізації цих завдань засобами мови C#.

Построить программу, которая вводит координаты точки  $(x, y)$  и определяет, попадает ли точка в заштрихованную область на рисунке, который соответствует Вашему варианту. Попадание на границу области считать попаданием в область.



```
using System;

class Program
{
    static void Main()
    {
        Console.WriteLine("Введите координаты точки (x, y):");
        double x = Convert.ToDouble(Console.ReadLine());
        double y = Convert.ToDouble(Console.ReadLine());

        if ((x >= -1 && x <= 0 && y >= 0 && y <= 1) ||
            (x >= 0 && x <= 1 && y >= -1 && y <= 0))
        {
            Console.WriteLine("Точка попадает в область.");
        }
        else
        {
            Console.WriteLine("Точка не попадает в область.");
        }
    }
}
```

```
Введите координаты точки (x, y):
0,5
-0,7
Точка попадает в область.
```

```
Консоль отладки Microsoft V × + ▾
Введите координаты точки (x, y):
5
5
Точка не попадает в область.
```

## Завдання 6\*

Виконати минуле завдання без використання оператора **if** у будь-яких формах

```
using System;

class Program
{
    static void Main()
    {
        Console.WriteLine("Введите координаты точки (x, y):");
        double x = Convert.ToDouble(Console.ReadLine());
        double y = Convert.ToDouble(Console.ReadLine());

        bool isInUpperLeft = (x >= -1 && x <= 0) && (y >= 0 && y <= 1);
        bool isInLowerRight = (x >= 0 && x <= 1) && (y >= -1 && y <= 0);

        bool isInShadedArea = isInUpperLeft || isInLowerRight;

        Console.WriteLine(isInShadedArea ? "Точка попадает в область." : "Точка не попадает в область.");
    }
}
```

## Завдання 7

Використовуючи завдання власного варіанта до лабораторної роботи 6 з електронних методичних вказівок з основ програмування та алгоритмічних мов створити проект з реалізації цих завдань засобами мови C#.

Для ряда, члены которого вычисляются по формуле, соответствующей Вашему индивидуальному заданию, подсчитать сумму членов ряда с точностью до 0.000001 и сумму первых 10 членов ряда. Если Вы считаете это необходимым, можете упростить или преобразовать выражение.

$$a_n = (-1)^n \frac{n+1}{n^3+2}$$

```
using System;

class Program
{
    static void Main()
    {
        double sum = 0;
        double epsilon = 0.000001;
        int n = 0;

        while (true)
        {
```

```

double term = (-1.0 * Math.Pow(-1, n) * (n + 1)) / (Math.Pow(n, 3) + 2);

if (Math.Abs(term) < epsilon)
    break;

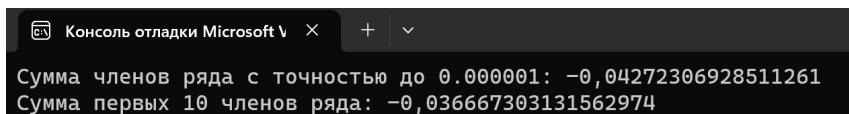
sum += term;
n++;
}

Console.WriteLine("Сумма членов ряда с точностью до 0.000001: " + sum);

// Вычисление суммы первых 10 членов ряда
double sumFirst10 = 0;
for (int i = 0; i < 10; i++)
{
    double term = (-1.0 * Math.Pow(-1, i) * (i + 1)) / (Math.Pow(i, 3) + 2);
    sumFirst10 += term;
}

Console.WriteLine("Сумма первых 10 членов ряда: " + sumFirst10);
}
}

```



Консоль отладки Microsoft V × + -

Сумма членов ряда с точностью до 0.000001: -0,04272306928511261  
 Сумма первых 10 членов ряда: -0,036667303131562974

## Завдання 8\*

Виконати минуле завдання, переробивши алгоритм вираховування суми ряду таким чином:

- похибка та кількість членів ряду мають вводитися як вхідні параметри у відповідні змінні;
- сума ряду вираховується або при досягненні заданої похибки, або при досягненні заданої кількості членів ряду із видачою відповідного повідомлення.

```

using System;

class Program
{
    static void Main()
    {
        Console.Write("Введите погрешность: ");
        double epsilon = Convert.ToDouble(Console.ReadLine());

        Console.Write("Введите максимальное количество членов ряда: ");
    }
}

```

```

int maxTerms = Convert.ToInt32(Console.ReadLine());

double sum = 0;

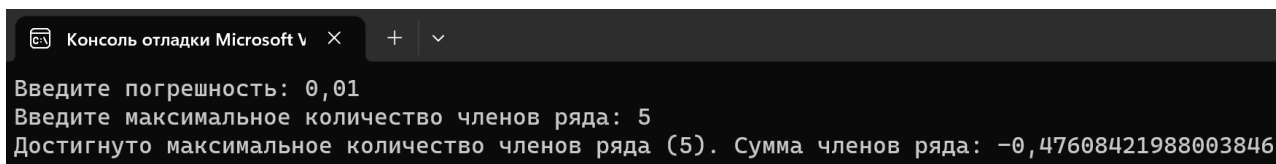
for (int n = 1; n <= maxTerms; n++)
{
    double term = Math.Pow(-1, n) * (n + 1) / (Math.Pow(n, 3) + 2);

    sum += term;

    if (Math.Abs(term) < epsilon)
    {
        Console.WriteLine($"Сумма членов ряда с достигнутой погрешностью
{epsilon}: " + sum);
        break;
    }

    if (n == maxTerms)
    {
        Console.WriteLine($"Достигнуто максимальное количество членов
ряда ({maxTerms}). Сумма членов ряда: " + sum);
        break;
    }
}
}
}

```



```

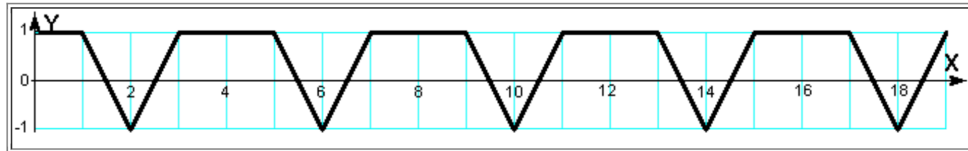
Консоль отладки Microsoft V  X  +  v
Введите погрешность: 0,01
Введите максимальное количество членов ряда: 5
Достигнуто максимальное количество членов ряда (5). Сумма членов ряда: -0,47608421988003846

```

- Завдання 9

Використовуючи завдання власного варіанта до лабораторної роботи 7 з електронних методичних вказівок з основ програмування та алгоритмічних мов створити проект з реалізації цих завдань засобами мови C#.

Для функции  $Y = f(X)$ , график которой приведен в Вашем индивидуальном задании, вывести на экран значения  $Y$  для  $X = 0, 0.25, 0.5, \dots 19.75$ . Необязательное для выполнения дополнительное требование - получить на экране график функции средствами текстового режима.



```
using System;
```

```
class Program
```

```
{
    static void Main(string[] args)
    {
        short n;    // параметр внешнего цикла
        double x, y; // абсцисса и ордината графика

        // внешний цикл - для 5 периодов
        for (n = 0; n < 5; n++)
        {
            // вывод заголовка таблицы
            Console.WriteLine("x | y");
            Console.WriteLine("-----|-----");

            // внутренний цикл для одного периода
            for (x = 0; x < 4; x += 0.25)
            {
                // 1-й отрезок
                if (x < 1) y = 2 * x - 1;
                // 2-й отрезок
                else if (x < 3)
                {
                    y = 1;
                }
                // 3-й отрезок
                else y = 7 - 2 * x;

                // вывод координат (x, y)
                Console.WriteLine($"{x + n * 4,5:F2} | {y,10:F7}");
            } // конец внутреннего цикла

            // пауза до команды оператора
            Console.WriteLine("Нажмите клавишу Enter...");
            Console.ReadLine();
        } // конец внешнего цикла

        return;
    } // конец метода Main
} // конец класса Program
```

```

C:\Users\dmitybirukov\source\...
10,00 | 1,00000000 |
10,25 | 1,00000000 |
10,50 | 1,00000000 |
10,75 | 1,00000000 |
11,00 | 1,00000000 |
11,25 | 0,50000000 |
11,50 | 0,00000000 |
11,75 | -0,50000000 |
Нажмите клавишу Enter...

  x   |   y   |
-----|-----|
12,00 | -1,00000000 |
12,25 | -0,50000000 |
12,50 | 0,00000000 |
12,75 | 0,50000000 |
13,00 | 1,00000000 |
13,25 | 1,00000000 |
13,50 | 1,00000000 |
13,75 | 1,00000000 |
14,00 | 1,00000000 |
14,25 | 1,00000000 |
14,50 | 1,00000000 |
14,75 | 1,00000000 |
15,00 | 1,00000000 |
15,25 | 0,50000000 |
15,50 | 0,00000000 |
15,75 | -0,50000000 |
Нажмите клавишу Enter...

```

## Завдання 10\*

Виконати минуле завдання, покращивши зовнішній вигляд побудованого графіку за допомогою символів псевдографіки \, /, —, \, /, та ін.) та налаштування кольорів консолі, наприклад, числа відобразити одним кольором, пояснюючий текст – іншим, графік – іншим від попередніх.

```

using System;
class Program
{
    static void Main(string[] args)
    {
        short n;    // параметр внешнего цикла
        double x, y; // абсцисса и ордината графика
        short h;    // позиция точки на экране

        // внешний цикл - для 5 периодов
        for (n = 0; n < 5; n++)
        {
            // вывод заголовка таблицы
            Console.WriteLine("| x | y |");
            Console.WriteLine("|-----|");

            // внутренний цикл для одного периода
            for (x = 0; x < 4; x += 0.25)
            {
                // 1-й отрезок
                if (x < 1) y = 2 * x - 1;
                // 2-й отрезок
                else if (x < 3)

```



```

    {
        y = 1;
    }
    // 3-й отрезок
    else y = 7 - 2 * x;

    // вывод строки таблицы
    Console.WriteLine($"{x + n * 4,5:F2} | {y,10:F7} |");

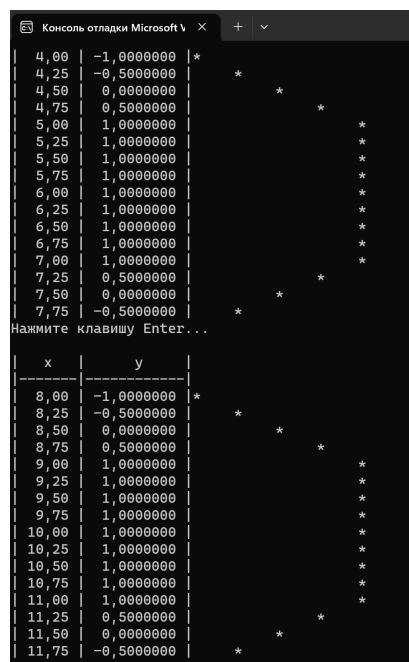
    // определение позиции точки
    h = (short)((y + 1) * 10);
    if (y - 1 - h * 10 > 0.5) h++;

    // вывод точки графика
    for (; h > 0; h--) Console.Write(" «");
    Console.WriteLine("*");
} // конец внутреннего цикла

// пауза до команды оператора
Console.WriteLine("Нажмите клавишу Enter...");
Console.ReadLine();
} // конец внешнего цикла

return;
} // конец метода Main
} // конец класса Program

```



Висновок: у даній лабораторній роботі було виконано 10 завдань різної складності. Були набуті практичні та теоритичні знання з таких тем :

- 1.Базові типи даних та введення-виведення
- 2.Арифметичні операції та математичні функції мови C#
- 3.Умовний оператор у мові C#
- 4.Оператори циклу в мові C#
- 5.Оператори циклу в мові C#
- 6.Робота з масивами

Також, під час виконання лабораторної роботи була розроблена програма на C#, призначена для обробки послідовності чисел, представленої у вигляді динамічної колекції `List<int>`.

Задачею програми було підрахувати кількість пар сусідніх елементів з однаковими значеннями в цій послідовності.

Програма успішно виконує поставлену задачу. Вона починає з генерації випадкової послідовності чисел, використовуючи генератор випадкових чисел. Потім вона проводить обхід цієї послідовності, обчислюючи середнє значення від'ємних елементів і усереднюючи значення позитивних елементів відповідно до умов завдання.

Для виділення пар з однаковими значеннями зеленим кольором програма перевіряє кожну пару сусідніх елементів і змінює колір тексту за необхідності.

Також програма коректно підраховує кількість пар сусідніх елементів з однаковими значеннями і виводить цю кількість на консоль. Отже, лабораторну роботу успішно виконано, і програму обробки послідовності чисел на C# створено і протестовано відповідно до поставлених завдань.

Також була виконана програма на C# ми побудували простий графік функції  $y = f(x)$  в консолі, використовуючи символи ASCII для подання точок на графіку. Програма визначає діапазон значень  $x$ , функцію  $f(x)$  (у цьому випадку синус), а потім створює графічне представлення цієї функції у текстовому вигляді.

Після виконання програми ми бачимо графік функції на консольному екрані. Цей приклад є простим і може бути розширений для більш складних функцій та креативного представлення графіка.

Основний висновок з цієї програми - ви можете використовувати мову програмування C# для створення текстових графіків функцій у консолі, що може бути корисним для візуалізації даних або навчання програмуванню.