

6 - ой семинар:

Задача 1: Создание строки из двумерного массива символов

```
public class Task1
{
    public static void Main(string[] args)
    {
        // Инициализация двумерного массива символов
        char[,] charArray = new char[,] { { 'a', 'b' }, { 'c', 'd' } };

        // Вызов метода для создания строки из 2D массива
        string result = CreateStringFrom2DArray(charArray);

        // Вывод результата
        Console.WriteLine(result);
    }

    // Метод для создания строки из двумерного массива символов
    public static string CreateStringFrom2DArray(char[,] array)
    {
        string result = "";

        // Цикл по каждому элементу в двумерном массиве
        for (int i = 0; i < array.GetLength(0); i++)
        {
            for (int j = 0; j < array.GetLength(1); j++)
            {
                // Добавление каждого символа в результирующую строку
                result += array[i, j];
            }
        }

        return result;
    }
}
```

Задача 2: Замена заглавных букв на строчные в строке

```
public class Task2
{
    public static void Main(string[] args)
    {
        // Входная строка со смешанными буквами обоих регистров
        string input = "aBcD1ef!-";

        // Преобразование всех заглавных букв в строчные
        string result = input.ToLower();
    }
}
```

```

        // Вывод результата
        Console.WriteLine(result);
    }
}

```

Задача 3: Проверка строки на палиндром

```

public class Task3
{
    public static void Main(string[] args)
    {
        // Входная строка для проверки
        string input = "шалаш";

        // Вызов метода для проверки, является ли строка палиндромом
        bool isPalindrome = IsPalindrome(input);

        // Вывод результата
        Console.WriteLine(isPalindrome ? "Да" : "Нет");
    }

    // Метод для проверки, является ли строка палиндромом
    public static bool IsPalindrome(string str)
    {
        // Нормализация строки путем удаления не буквенно-цифровых
        // символов и приведения к нижнему регистру
        string normalized = new
        string(str.Where(char.IsLetterOrDigit).ToArray()).ToLower();

        // Сравнение строки с ее перевернутым вариантом
        return normalized.SequenceEqual(normalized.Reverse());
    }
}

```

Задача 4 (необязательная): Обратный порядок слов в строке

```

public class Task4
{
    public static void Main(string[] args)
    {
        // Входная строка со словами, разделенными пробелами
    }
}

```

```
string input = "Hello my world";

// Вызов метода для обращения порядка слов в строке
string result = ReverseWords(input);

// Вывод результата
Console.WriteLine(result);
}

// Метод для обращения порядка слов в строке
public static string ReverseWords(string str)
{
    // Разделение строки на слова
    string[] words = str.Split(' ');

    // Обращение порядка слов
    Array.Reverse(words);

    // Соединение слов обратно в строку с пробелами
    return string.Join(" ", words);
}
}
```