

LINQ

Задача 1.

Пользователь вводит количество чисел, после чего генерируется массив, содержащий нужное количество натуральных чисел от 1 до 9 случайным образом. Напишите метод, который вернет массив всех различных чисел, присутствующих в сгенерированном массиве, в порядке возрастания.

Задача 2.

Пользователь вводит построчно точки, заданные координатами X и Y, разделенные пробелом (конец ввода — Enter). Реализуйте метод ParsePoints, принимающий список введенных строк и возвращающий список точек, в одно LINQ-выражение.

```
public class Point
{
    public int X, Y;

    public Point (int x, int y)
    {
        X = x;
        Y = y;
    }
}
```

Задача 3.

Вам дан список всех академических групп. Нужно получить список всех студентов всех групп. Академическая группа определена так:

```
public class Group
{
    public List<string> Students = new List<string>();
}
```

Без использования LINQ решение могло бы выглядеть так:

```
var allStudents = new List<string>();
foreach (var group in groups)
{
    foreach (var student in group.Students)
    {
        allStudents.Add(student);
    }
}
return allStudents;
```

Напишите решение этой задачи с помощью LINQ в одно выражение.

Задача 4.

Рассмотрим классы `Photo` и `Pixel` из проекта `PhotoEnhancer`.

Создать метод расширения класса `Photo` — метод `GetNeighbours`, возвращающий список пикселей всех соседей заданной точки в изображении (включая заданную точку).

Задача 5.

Текст находится в текстовом файле. Вам нужно составить лексикографически упорядоченный список всех встречающихся в этом тексте слов и вывести этот список на консоль.

Задача 6.

Текст находится в текстовом файле. Вам нужно составить упорядоченный сначала по возрастанию длины слова, а потом лексикографически, список всех встречающихся в этом тексте слов и вывести этот список на консоль.

Задача 7.

Текст находится в текстовом файле. Нужно найти самое длинное слово в этом тексте, а из всех самых длинных — лексикографически первое слово. Задачу решить в одно LINQ-выражение, не используя методы сортировки LINQ.

Задача 8.

В академических группах задачи 3 написать методы, подсчитывающие общее число студентов во всех группах и среднее количество студентов. Каждый метод должен содержать одно LINQ-выражение. Методы не должны вызывать друг друга.

Задача 9.

Написать методы, которые определяют по заданному целочисленному массиву

- a) Являются ли все элементы массива положительными числами?
- b) Содержит ли массив число 0?

Каждый метод должен быть написан в одно LINQ-выражение.

Задача 10.

Написать метод, находящий среднее геометрическое заданного непустого списка действительных чисел. Метод не должен содержать циклов. Для пустого списка метод должен пробрасывать исключение `ArgumentException`.

Задача 11.

Текст находится в текстовом файле. Составьте словарь частот слов, начинающихся на определенную букву алфавита (т. е. отношений числа слов, начинающихся на букву, к общему числу слов в тексте). Словарь распечатайте частоты в таблицу с точностью до 4 знаков после запятой в виде

А: 0,1673

Б: 0,0228

и т. д.

Если на какую-то букву в тексте не начинается ни одного слова, соответствующую частоту, равную 0, печатать не надо. Метод получения словаря не должен содержать циклов.

Задача 12.

Очень старая процедурная программа хранит данные о пользователях сервиса в двух отдельных массивах. В одном хранятся логины пользователей, а во втором, по соответствующим индексам, — хэш-коды их паролей.

Объектно-ориентированный программист Олег решил инкапсулировать данные о пользователях, для чего написал класс User:

```
public class User
{
    public string Login;
    private int passwordHashCode;

    public User (string login, int hashCode)
    {
        Login = login;
        passwordHashCode = hashCode;
    }

    public bool IsPasswordCorrect(string password)
    {
        return password.GetHashCode() == passwordHashCode;
    }
}
```

Помогите Олегу — напишите метод, который из двух массивов сделает словарь пользователей (ключ — логин, значение — пользователь).