**Выполнили студент 215 группы**

**Марков К.А.**

**Практическое занятие 20**

**Тема: «ТИПЫ ПОТОКОВ. УПРАВЛЕНИЕ ПОТОКАМИ»**

**Цель лабораторной работы:** освоить принципы создания и управления потоками.

**Задание:**

1. Создайте консольное приложение.

2. Разработайте классы для решения задачи в соответствии с индивидуальным вариантом с использованием пула потоков. В каждом задании необходимо разработать коллекцию элементов заданного типа и класс для управления коллекцией, осуществляющий обработку элемента с использованием пула потоков.

3. В алгоритме решения задачи предусмотрите задержку алгоритма с использованием метода Thread.Sleep( ).

Вариант 11: Метод находит символ строки, который встречается максимальное количество раз.

public class CharCounter

{

private string inputString; // Строка, для которой будет выполняться подсчет символов

// Конструктор класса CharCounter

public CharCounter(string input)

{

inputString = input; // Инициализация входной строки

}

// Метод для нахождения символа, который встречается максимальное количество раз в строке

public char FindMostFrequentChar()

{

// Словарь для подсчета количества вхождений каждого символа

Dictionary<char, int> charCount = new Dictionary<char, int>();

// Подсчет количества встреч каждого символа в строке

foreach (char c in inputString)

{

if (charCount.ContainsKey(c))

charCount[c]++; // Если символ уже встречался, увеличиваем счетчик

else

charCount[c] = 1; // Если символ встречается впервые, инициализируем счетчик

}

// Нахождение символа с максимальным количеством вхождений

int maxCount = 0; // Максимальное количество вхождений

char mostFrequentChar = '\0'; // Символ с максимальным количеством вхождений

foreach (var pair in charCount)

{

if (pair.Value > maxCount)

{

maxCount = pair.Value; // Обновляем максимальное количество вхождений

mostFrequentChar = pair.Key; // Запоминаем символ с максимальным количеством вхождений

}

}

return mostFrequentChar; // Возвращаем символ с максимальным количеством вхождений

}

}

public class Program

{

public static void Main(string[] args)

{

string input;

do

{

Console.Write("Введите строку: ");

input = Console.ReadLine();

if (string.IsNullOrEmpty(input))

{

Console.WriteLine("Строка не может быть пустой. Пожалуйста, введите ее снова.");

}

} while (string.IsNullOrEmpty(input));

CharCounter charCounter = new CharCounter(input);

// Создаем задачу, которая будет выполнять поиск символа с максимальным количеством вхождений

ThreadPool.QueueUserWorkItem(state =>

{

char mostFrequentChar = charCounter.FindMostFrequentChar();

Console.WriteLine($"Символ, встречающийся максимальное количество раз: {mostFrequentChar}");

});

// Имитируем выполнение других задач, пока выполняется поиск символа

for (int i = 0; i < 5; i++)

{

Console.WriteLine($"Выполнение другой задачи {i}");

Thread.Sleep(1000); // Задержка в 1 секунду

}

Console.ReadLine();

}

}

**Вывод:** освоил принципы создания и управления потоками.