

## ТЕМА 3 РЕГУЛЯРНЫЕ ВЫРАЖЕНИЯ В ЯЗЫКЕ C#

### 3.1 Цель лабораторной работы

Получить практические навыки по программированию регулярных выражений в языке C#.

### 3.2 Методические указания к лабораторной работе

Изучить материал 5 и 6 лекций дисциплины.

Изучить материал, рекомендуемый в силлабусе: стр. 355-365 [2]. – сокращенный вариант представлен:

Стандартный класс `String` позволяет выполнять над строками различные операции, в том числе поиск, замену, вставку и удаление подстрок. Тем не менее, есть классы задач по обработке символьной информации, где стандартных возможностей явно не хватает. Чтобы облегчить решение подобных задач, в `Net Framework` встроен более мощный аппарат работы со строками, основанный на регулярных выражениях. Специальное пространство имен `RegularExpression`, содержит набор классов, обеспечивающих работу с регулярными выражениями. Полное описание, как теоретических основ, так и практических особенностей применения этого аппарата в C#, требует отдельной книги. Придется ограничиться конспектом лекций введением в эту интересную область работы со строками, не рассматривая подробно все классы, входящие в пространство имен `RegularExpression`.

### 3.3 Домашнее задание на лабораторную работу

Считать текст из файла и вывести его на экран монитора только предложения, состоящие из заданного количества слов (любых наборов символов между пробелами).

### 3.4 Индивидуальные задания для СРС

Дополнение 1. В некоторых индивидуальных заданиях необходимо выполнить анализ информации файла телефонных переговоров клиента, который среди прочей информации содержит записи об абонентах, включающих следующие сведения:

- номер абонента, с которым выполнялось соединение;
- дату и время соединения;
- продолжительность соединения;
- стоимость переговоров.

Дополнение 2. В некоторых индивидуальных заданиях необходимо выполнить анализ кода программы на наличие в нем различных элементов языка C#. Код программы должен быть представлен обычным файлом с

расширением «.cs»), в который Вы записываете необходимую информацию в соответствии с Вашим индивидуальным заданием.

Дополнение 3. В некоторых индивидуальных заданиях необходимо выполнить анализ некоторого текста. Текст, соответствующий Вашему заданию, должен быть помещен в текстовый файл с расширением «.txt» или «.doc».

3.4.1 В файле с телефонными переговорами клиента определить по какому номеру выполнено максимальное количество звонков.

3.4.2 Выполнить анализ кода программы на наличие в нем всех циклов for языка C#. Напечатать результаты анализа на экране монитора.

3.4.3 Считать текст из файла и вывести на экран монитора строку, содержащую максимальное количество знаков пунктуации.

3.4.4 В файле с телефонными переговорами клиента выполнить сортировку переговоров по их стоимости.

3.4.5 Выполнить анализ кода программы на наличие в нем комментариев языка C#. Напечатать результаты анализа на экране монитора.

3.4.6 Считать текст из файла и вывести его на экран монитора, заменив цифры словами, например, «0» на слово «ноль»; «1» на слово «один» и т.д.

3.4.7 В файле с телефонными переговорами клиента определить по какому номеру выполнено максимальное количество звонков.

3.4.8 Выполнить анализ кода программы на наличие в нем всех операторов присваивания языка C# (не учитывать записи типа for (i=0; i<=10; i++) ... ). Напечатать результаты анализа на экране монитора.

3.4.9 Считать текст из файла и вывести на экран монитора строку, содержащую максимальное количество чисел (не цифр).

3.4.10 В файле с телефонными переговорами клиента определить по какому номеру был самый продолжительный по времени разговор.

3.4.11 Выполнить анализ кода программы на наличие в нем всех операторов вывода информации на экран (консоль) языка C#. Напечатать результаты анализа на экране монитора.

3.4.12 Считать текст из файла и вывести его на экран монитора только цитаты текста, то есть предложения, заключенные в кавычки.

3.4.13 В файле с телефонными переговорами клиента определить по какому номеру выполнены подряд (2 и более соединений) звонки.

3.4.14 Выполнить анализ кода программы на наличие в нем всех операторов ввода информации с клавиатуры (консоли) языка C#. Напечатать результаты анализа на экране монитора.

3.4.15 Считать текст из файла и вывести на экран монитора строку, содержащую максимальное количество повторяющихся слов.

3.4.16 В файле с телефонными переговорами клиента определить по какому номеру выполнена максимальная непрерывная последовательность звонков.

3.4.17 Выполнить анализ кода программы на наличие в нем всех операторов условного перехода языка C#. Напечатать результаты анализа на экране монитора.

3.4.18 Считать текст из файла и вывести его на экран монитора только предложения, содержащие информацию о 1000 и более «у.е.».

3.4.19 В файле с телефонными переговорами клиента определить все номера, время связи с которыми было менее 5 секунд.

3.4.20 Выполнить анализ кода программы на наличие в нем всех циклов с условием, но не операторов if языка C#. Напечатать результаты анализа на экране монитора.

### **3.5 Контрольные вопросы для защиты отчета на СРСП**

3.5.1 Понятие регулярного выражения. Примеры.

3.5.2 Основные задачи, решаемые с помощью регулярных выражений.

3.5.3 Символика языка регулярных выражений. Пример.

3.5.4 Понятие повторителей. Примеры.

3.5.5 Понятие метасимволов. Примеры.

3.5.6 Понятие заменителей или «Классов символов», Примеры.

3.5.7 Специальные (управляющие) символы. Примеры.

3.5.8 Методы регулярных выражений. Примеры.