

Compiler Construction

Mid Lab

Name: Muhammad Salman Umer

Reg No: FA20-BCS-013

Submitted To: Syed Bilal Haider

Date: 26-10-2023

Qs: 1 Describe functioning of regex C# library , giv examples of patterns,seperators and anchors e.t.c.

The System in C#Text.Classes for using regular expressions are available in the RegularExpressions namespace. Regular expressions are effective tools for searching and matching patterns within strings. Regular expressions are often abbreviated as regex or regexp, are powerful search patterns used for string manipulation and matching in text. They consist of a sequence of characters that define a search pattern.Here are several patterns, separators, and anchor samples, as well as an explanation of how the library works:

**1. Creating Objects with Regex:**

By instantiating the Regex class and passing it a pattern as a parameter, you can produce a Regex object.

**2. Patterns:**

\d: Matches any digit (equivalent to [0-9]).

\w: Matches any word character (alphanumeric character plus underscore).

.: Matches any character except for a newline.

[]: Matches any one of the characters inside the brackets. For example, [aeiou] matches any vowel.

+: Matches one or more occurrences of the preceding pattern.

\*: Matches zero or more occurrences of the preceding pattern.

?: Matches zero or one occurrence of the preceding pattern.

^: Matches the start of a string.

$: Matches the end of a string.

**3. Anchors:**

^: Anchors the regex at the start of the string. For example, ^Hello matches a string that starts with "Hello".

$: Anchors the regex at the end of the string. For example, World$ matches a string that ends with "World".

**4. Separators:**

\s: Matches any whitespace character (space, tab, newline).

\S: Matches any non-whitespace character.

\b: Matches a word boundary.

\B: Matches a non-word boundary.