Principles of Programming

[MCS 1205]

Assignment 01 – Regular Expressions for Scanners

1. Objective

This assignment focuses on practicing and evaluating the design of Regular Expressions for the Lexical Analysis or Scanner building stage of the language implementation process.

2. Introduction

In the Principles of Programming language analysis course, we have learnt that there are two primary steps for recognising and analysing languages – Lexical Analysis (Scanner) and Syntax Analysis (Parser). The Scanner is primarily to recognise Lexemes and group them into tokens for the syntax analysis process to apply the rules of a grammar.

3. Duration

The duration for the assignment is 08 days and should be completed and submitted individually. **Submission Deadline – 19th May 2024**.

4. Submission Guidelines

The assignment submission should comply with the following criteria.

1. The regular expressions for the given questions/scenarios should be clearly documented.
2. The Jupyter file containing the implementation of the given tasks should be attached with the submission.
    a. The Flex program used to implement the solution to each task should be compiled without errors.
    b. Compile the Flex generated output using a C language compiler to result in a executable that can be used as a scanner.
    c. Run negative and positive examples of inputs given in the tasks to the scanner and

confirm the accuracy of the proposed regular expression.

3. A self-contained and commented Jupyter file with Flex based regular expressions, compiling and testing would be accepted as a complete submission.

5. Assignment Outline

Task 1 – Recognise a Special String Label

Pattern Description: Write Flex rules (regular expressions) that matches identifiers which can start only with an uppercase letter or *(Asterisk) and can contain letters, digits, or underscores thereafter.

Compatible input strings that should be recognised by the scanner: Name23, *aGe9_

Incompatible input strings that should be rejected by the scanner: 54Saman, saman54

Task 2 – Recognise signed floating point numbers

Pattern Description: Write Flex rules (regular expressions) that matches and captures floating point numbers with a negative or positive sign at the start. The sign is mandatory.

Compatible input strings that should be recognised by the scanner: +9.81, -456.98

Incompatible input strings that should be rejected by the scanner: 456.45, +56.34.22

Task 3 – Recognise single and multi-line comments

Pattern Description: Write a regular expression that matches single line comments (// ....) and multiline comments (/*.. */) in a language similar to C.

Compatible input strings that should be recognised by the scanner:

// This is a comment

/* This is a multi line comment */

Incompatible input strings that should be rejected by the scanner:

/ This is a wrong single line

/* this is a wrong multiline comment //

Task 4 – Recognise email addresses

Pattern Description: Write a Flex rule that matches valid email addresses consisting of a username, an "@" symbol, and a domain name. The username can contain letters, digits, underscores, dots, and hyphens. The domain name can contain letters, digits, and hyphens.

Compatible input strings that should be recognised by the scanner:
nicola.tesla@dcelectic.com
nicola_tesla@dc-electic.com
nicola.tesla1856@dc-electic.com

Incompatible input strings that should be rejected by the scanner:
nicola.tesla1856@dc*electic.com

Task 5 – Recognise Hexadecimal Numbers

Pattern Description: Write a Flex rule that matches hexadecimal numbers starting with "0x" or "0X" and followed by a sequence of hexadecimal digits (0-9, A-F, a-f).

Compatible input strings that should be recognised by the scanner: 0X3F, 0Xaf, 0x13, 0x9ADF

Incompatible input strings that should be rejected by the scanner: 0X0GAf, 0x87AZ

Task 6 – Recognise Date Formats

Pattern Description: Write a Flex rule that identifies dates in the following two formats. DD-MM-YYYY (e.g., 26-04-2024)

Positive input strings that should be recognised by the scanner:

16-06-1970
31-12-1999
01-12-0999

Incompatible input strings that should be rejected by the scanner:

01-13-2000
01-00-2000
00-12-2000
32-12-2000

Task 7 – Recognise HTML formats

Pattern Description: Write Flex rules to identify and capture HTML opening and closing tags with attributes.

Compatible input strings that should be recognised by the scanner:

```
<HTML>
<script>
<div class="container">
<script>
  const myString = "<script></script>"
</script>
<div class="container">
  <!-- <img src="cat.jpg" alt="big cat" > -->
</div>
</HTML>
```

Task 8– Recognise URLs

Pattern Description: Write a Flex rule that matches protocol (http, https), domain name and optional port number, and path.

Compatible input strings that should be recognised by the scanner:

```
http://google.com
https://google.com:6060
https://google.com:6060/gemini
```

---------- [END] ----------