Advanced Topics – Compilers and Interpreters

Lab: JFlex demonstration

In this exercise we will study regular expressions and use the Scanner generator tool JFlex to generate a Scanner capable of tokenizing input files according to the specifications in the .flex file defined using regular expressions.

JFlex is a utility that uses regular expressions, converting them to NFA then DFA then to a table-driven implementation (Java code) for tokenizing text input files.

Exercise: Regular Expressions

Study regular expressions focusing on using regular expressions to identify patterns in data for ex. Most US Phone numbers match the pattern below:

```
\+?1?\(?\d{3}\)?[-.]?\d{3}[.-]?\d{4}
```

Your first task is to design five interesting regular expressions to match and tokenize a data set. Some suggestions for data sets are:

- Amazon Product Data
- Social media feeds
- News feeds
- Census data
- Pollution data
- Marine life data

These data sets may be text, json or rss. Once you identify the data set you want to use you can start looking at patterns ex. Email addresses, urls, hashtags, prices, telephone numbers among other things in the data and then design the regular expressions to recognize them.

Exercise: The Flex File

Next you will use the template ScannerAbb.flex provided to you to create your own flex file. The flex file is the input to the Scanner generator. Make sure to define at least a few (3+) Pattern definitions or macros:

```
/**
 * Pattern definitions
 */
Letter          = [a-zA-Z] | "_"
Digit           = \d
Identifier      = {Letter}({Letter} | {Digit})*
```

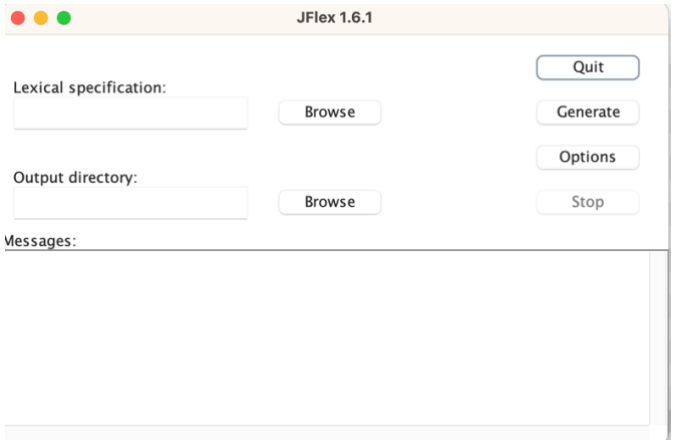

and then define 5-15 lexical rules (actions):

```
/**
 * lexical rules
 */
{Identifier}     {return "<Identifier:" + yytext()+">";}
```

Exercise: Generating the Scanner

Once you have your flex file ready download and execute the jflex-1.6.1.jar file by following the link on Schoology in the Homework and Lab Assignments folder under PROJ: JFlex or by visiting the website https://jflex.de/
Once you unzip the folder launch JFlex by navigating to the lib folder inside jflex-1.6.1 or jflex-1.9.1 depending on your installation.

You should see the window below:



Click the browse button next to Lexical specification and provide you .flex file as input.
Next specify the output directory where you want your generated Scanner.java file to be placed. Then click the Generate button. You should see the parsing of the file, generation of the NFA, DFA, table with the number of states etc if there are no errors in your flex file and the Scanner.java file will be created in the folder you specified. If there are any errors fix them in your flex file and repeat the process. Once your Scanner is generated you are done.

Exercise: Demonstrate the generated Scanner

Finally, you are going to back up your Scanner lab then replace your handwritten Scanner with the autogenerated Scanner in the lab and replace your input file (scannerTest.txt or ScannerTestAdvanced.txt) in the tester with your data set (Amazon product data) and run the tester to see the tokenized output.

Exercise: Presentation in class

On a pre-decided day, you along with your partner(s) will present your study, analysis of regular expressions, flex file and Scanner demo on the data set of your choice for a grade. You will be graded based on your choice of data set, number and complexity of regular expressions, pattern definitions, lexical rules, and presentation skills. The presentation should have approximately 6-8 slides. Intro, reg ex, flex file, data set, citations and conclusion.

