**John Scales**

**CPSC 323**

**Project 1**

**6 October 2019**

**CS323 Documentation**

About 2-3 pages

1. **Problem Statement**

Our job is to make a lexical analyzer for the RAT19F language that we were given the rules for in class. We will need finite state machines to handles the RAT19F source code.

*Our main program will test the lexer we made. The program should read a file containing the source code of Rat19F to generate tokens and write out the results to an output file.*

*Make sure that you print both, the tokens and lexemes.*

*Basically, our main program should work as follows*

*while not finished (i.e. not end of the source file) do*

*call the lexer for a token*

*print the token and lexeme*

*endwhile*

1. **How to use your program**

*Extract contents of zip, open the unzipped directory in terminal, execute test by typing ./test in terminal. The program will read sample.rat19 and output the results to the console and to the file named output.txt*

1. **Design of your program**

*< write major components of your program. Also, data structures you are utilizing, particular algorithms you have chosen etc. >*

*One major component is the int/real FSM which uses a DFSM to decide if the input is valid, invalid, and real, or an integer.*

*RE int: (digit)\**

*RE real: (digit)(digit)\*.(digit)\**

*Another major component is the in\_array function to see if a given key is in an array. It’s a basic sub string matching algorithm.*

*We used mainly arrays to store the chars, but we also used strings to hold the completed input and print them to the console.*

*Inside the main while loop we have the arrays of separators and keywords for matching.*

*We then have the first if statement for identifying keywords vs identifiers.*

*RE id: alpha(\_|digit|alpha)\**

*The second if statement will check if the input is a digit and if it is it will check if its a real or integer.*

*The third if statement will check for separators and we specifically put this before the operators because had we not, we would have to do more work if the operators if statement encountered an asterisk than if the separators one did. Last we have the operators FSM which is a very basic string matching algorithm that comes after the separators for the reason stated earlier.*

1. **Any Limitation**

*Limitations are:*

*identifiers can only be 100 chars long*

*integers, and reals are only as long as a string*

*other limitations are built into the syntax such as keyword, operator, and separator length.*

1. **Any shortcomings**

*Not every FSM has an array of state numbers even though they are technically going through various states when we do the tests.*