Indian Institute of Information Technology Guwahati

Assignment 1
CS321: Compilers Lab
3rd August, 2017

Write a C program to design a lexical analyzer as discussed in class. The details are mentioned below:

The input strings must be in the following format:

variable = expression having *variables*, *number* and *binary operators.*

Here *variable* is like the variables declared in programming languages, *binary operator* means **+, - ,*** and **/**. **Number** means floating point number.

Some example strings are:

- **position=item+value*30.2**
- **interest=principal*year*rate/100**
- **new_value=old_value+65.0/new_value*34.2**

Your program must take a line (as mentioned above) as input and display all the tokens along with the final content of symbol table. The format of token and symbol table is discussed in class.

The rules for defining different tokens are:

1. **id:** any variable is an id. A variable name starts with a letter or '_' and then combination of any letter, digit or '_'.
2. **num**: supports only floating point numbers. e.g. 70.50, 0.45, 9.8 etc.
3. operators: all the operators are token itself.

Steps to do the task (as discussed in class):

1. Write the regular expression for each token pattern.
2. Design the DFA from the regular expressions.
3. Implement the DFA as discussed in class.

Example input/output

Input: position=first_index+item*position+60.5

Output:

Tokens: <id,1>, <=>, <id, 2>, <+>, <id, 3>, <*>, <id,1>, <+>, <num, 4>

Symbol table:

1	Position	Id
2	First_index	Id
3	item	Id
4	60.5	num