Compiler Construction

Assignment # 02 – Lexical Scanner

By Mr. Usman Wajid

1. To build **your programming language**, define the following,
    a. Rules for identifier name (Regular Expression)
    b. Reserve words including data types such as int, float, string etc
    c. Operators
    d. Parathesis
    e. Symbol used for end statement (use any symbol other than ; (semi-colon))

2. Draw a single **DFA** for your own language (use JFLAP). See "CC lecture 08" (page 3) for the DFA designed for the simple scanner

3. Write a lexical analyzer of Scanner **C++ code** for a **DFA** designed in step 2. That is capable of the following,
    a. The scanner should take the source code from a text file and store it in an input tape, say a character array.
    b. The input tape (char array) should have two pointers, i.e, start_point and end_point, you can declare both of them as integer variables. The start_point will keep track of starting symbol index from the input tape. Similarly, the end_point will keep track of the ending symbol index of the character that has been read so far from the input tape.
    c. Exclude spaces, tabs or line breaks
    d. Assign tokens for each valid lexeme (words)
    e. Store these tokens in a symbol table that could be any database, simple excel-sheet or even a text file

4. WHAT SHOULD YOU SUBMIT?
    a. JFLAP DFA design (Screen shot only)
    b. Your Scanner code file (Simple C language code is recommended, such as scanner.c)
    c. Sample output (screen shot only)

NOTE:

There is a **zero-tolerance policy on plagiarized assignments**. I don't expect you to know everything. If you are unsure of something, look it up. Try and learn it. Just do your best with what you have been taught in class. If you have achieved what the assignment was supposed to test you on, then you did well. You don't have to follow the task instructions exactly the same way as they are written. Don't be afraid to be creative, but Don't lose sight of the objective.