Project #1

CS 3510 – Spring 2015

Nathan Harris and Jake Prem

I. Requirements: Create a Scanner for the C- language

II. Design: Our first step in the design process was to create our regular expressions and DFAs. At first, we weren’t sure exactly what aspects of the language we would have to make regular expressions and DFAs for, but we settled with identifiers, numbers, and comments. The other special characters such as <=, ==, and >= were implemented in a similar way to comments.

III. Implementation: The most difficult part of implementation was figuring out a way to generate the correct tokens in the case of tokens such as < and <=. We settled on intermediate states such as the LESSTHAN state in which we would have a “<” character but would not save it as a LESSTHAN token unless the next character was not an equals sign. In this case, we used our PushbackReader and unread the current character, allowing us to continue in the same spot for the next token.

IV. Testing: We tested our program using a simple quicksort algorithm and the example program from the text on pg 27.

V. Summary/Conclusion: The program appears to function properly. It gives the correct answer for every input.

VI. Lessons Learned: The hardest part of this assignment was probably the preparation stage and trying to figure out exactly what had to be done in this project. We spent a couple hours just trying to figure out what regular expressions we would need and what DFAs we would need to create. Once we figured all of that out, the implementation was rather straightforward.

.

Our code compiled and ran properly, and produced the correct output