Compiler Design Lab (CS 511)

Autumn 2021

Assignment 2

Design a token recognizer for a C/C++ like program segment using Lex.

Output tokenized string(s) as shown below.  
Your designed recognizer should at least handle the following tokens:

1. keyw: int, double, for, while, if, then, else, do  
2. id: variable names  
3. delim: `;' and `}' to understand end-of-statement and end-of-block respectively. Consider `,' also as a delimiter.  
4. rel-op: { <, >, ==, <=, >=, != }  
5. assign\_op: =  
6. arith\_op: +, -, /, \*  
7. num: 123.12, 123, 0.12 etc.

\*\* The analyzer should ignore redundant spaces, tabs, new lines, and comments.

Sample Input Output

**Input:**  
int a, b, c;  
int sum1, sum2, sum;  
sum1 = a +b; /\* get two sums \*/  
sum2= b+ c;  
sum = sum1+sum2+100;

**Output:**  
<keyw><id><delim><id><delim><id><delim>  
<keyw><id><delim><id><delim><id><delim>  
<id><assign\_op><id><arith\_op><id><delim>  
<id><assign\_op><id><arith\_op><id><delim>  
<id><assign\_op><id><arith\_op><id><arith\_op><num><delim>