John Lawler

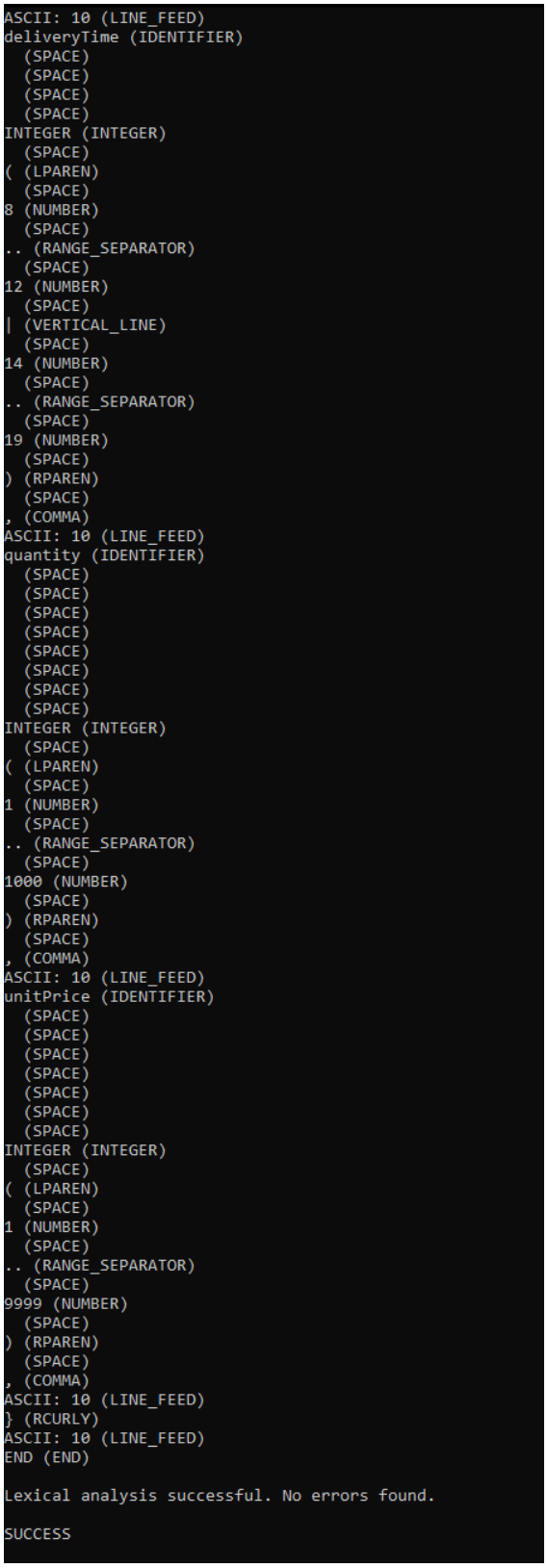
Professor Neeraj Gupta

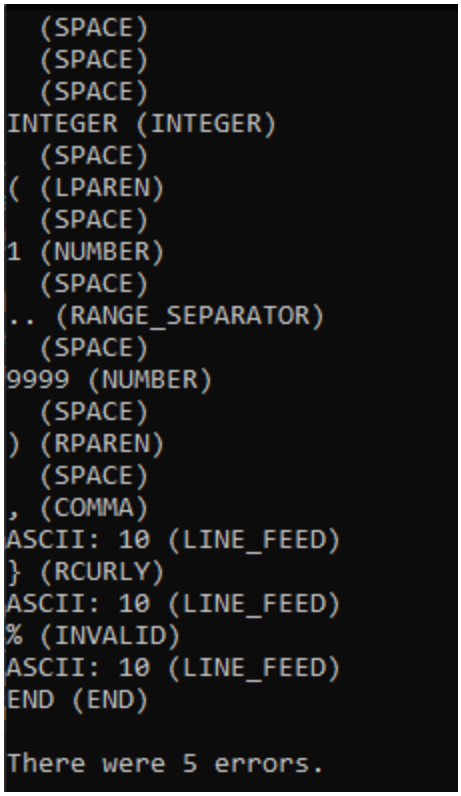
CS 4386.001 Compiler Design

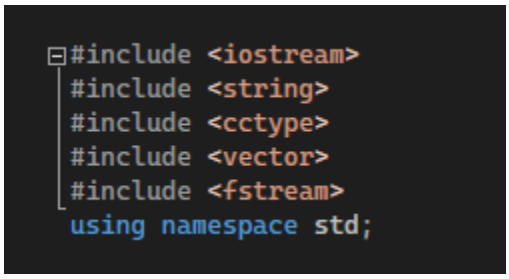
February 19th, 2023

1. The lexer is written in C++. It is included in the submission.
2. Included.

3.


ASCII: 10 (LINE_FEED)
deliveryTime (IDENTIFIER)
(SPACE)
(SPACE)
(SPACE)
(SPACE)
INTEGER (INTEGER)
(SPACE)
((LPAREN)
(SPACE)
8 (NUMBER)
(SPACE)
.. (RANGE_SEPARATOR)
(SPACE)
12 (NUMBER)
(SPACE)
| (VERTICAL_LINE)
(SPACE)
14 (NUMBER)
(SPACE)
.. (RANGE_SEPARATOR)
(SPACE)
19 (NUMBER)
(SPACE)
) (RPAREN)
(SPACE)
, (COMMA)
ASCII: 10 (LINE_FEED)
quantity (IDENTIFIER)
(SPACE)
(SPACE)
(SPACE)
(SPACE)
(SPACE)
(SPACE)
(SPACE)
(SPACE)
INTEGER (INTEGER)
(SPACE)
((LPAREN)
(SPACE)
1 (NUMBER)
(SPACE)
.. (RANGE_SEPARATOR)
(SPACE)
1000 (NUMBER)
(SPACE)
) (RPAREN)
(SPACE)
, (COMMA)
ASCII: 10 (LINE_FEED)
unitPrice (IDENTIFIER)
(SPACE)
(SPACE)
(SPACE)
(SPACE)
(SPACE)
(SPACE)
(SPACE)
INTEGER (INTEGER)
(SPACE)
((LPAREN)
(SPACE)
1 (NUMBER)
(SPACE)
.. (RANGE_SEPARATOR)
(SPACE)
9999 (NUMBER)
(SPACE)
) (RPAREN)
(SPACE)
, (COMMA)
ASCII: 10 (LINE_FEED)
} (RCURLY)
ASCII: 10 (LINE_FEED)
END (END)
Lexical analysis successful. No errors found.
SUCCESS

4. 

5. 

    a. iostream for cout/cin
    b. string for strings and string operations
    c. cctype to parse upper and lower case
    d. vector for a vector that I used
    e. fstream to open a prewritten file/write out to a file

6. The code (as far as the lexical analysis, there are some basic file operations and output coding used) is based on a DFA. This DFA can be found in the function lex since it seemed overkill to create regular expressions with this criteria. There are three main

sections to the DFA: alphabetical, numerical, and special characters. Once it decides which category the character falls into, the program then decides appropriately how to categorize it, and adds it to the vector of token types and lexemes.