CS323 Documentation
About 2-3 pages

1. **Problem Statement**

The second assignment is to write a syntax analyzer. You may use any top-down parser such as a RDP, a predictive recursive descent parser or a table driven predictive parser.

Hence, your assignment consists of the following tasks:

1. Rewrite the grammar Rat18S to remove any left recursion

(Also, use left factorization if necessary)

*2. Use the **lexer()** generated in the assignment 1 to get the tokens*

*3. **The parser should print to an output file the <u>tokens, lexemes</u> and the <u>production rules used;</u>***

*That is, <u>**first, write the token and lexeme found**</u>*

*Then, <u>**print out all productions rules**</u> used for analyzing this token*

*4. **Error handling:** if a syntax error occurs, your parser should generate a meaningful error message, such as token, lexeme, line number, and error type etc.*

Then, your program may exit or you may continue for further analysis.

2. **How to use your program**

<write detailed steps how to execute your program>

- *Run Syntax.exe*
- *Input test case text file, it can be named whatever*
- *Insert output file name*

3. **Design of your program**

The major components of the program:

Lexical.h : In the lexical function, it decides what token it is. If it's a separator, operator, real integer, integer or identifier. Then passes it to the correct function.

Syntax.h : In the Syntax header file, it parces through the Rat18s requirements.

Main.cpp : reads in the input file and creates an output file. It gets the line from the input file and then sends it the set_token function.

4. **Any Limitation**
   <All features are running according to the assignment but you limit your program due to resource limitations, such as
   *Maximum number of lines in the source code, size of the identifier, integer etc. **Say 'None' if there is no limitation**>*

   None.

5. **Any shortcomings**
   *<Anything you could NOT implement although that is required by the Assignment. **Say 'None' if there is no shortcoming**>*

   Nested Parenthesis does not work and Multiple line comments does not work.