

# COVER PAGE

## CS323 Programming Assignments

Name [ 1.Donghao Feng ], (MW 1:00 pm - 2:15 pm )

[ 2. Yong Kim ], (MW 5:00 pm - 6:45 pm )

[ 3. Haojie Pan ], (Tue Thurs 4:00 - 5:15 pm )

Assignment Number [2]

Due Dates: Softcopy [05/01/2019], Hardcopy [05/01/2019]

Turn-In Dates: Softcopy [4/28/2019], Hardcopy [4/28/2019]

Executable FileName [TDP]

FileName : TDP\_YKim\_iter2.cpp

Lab Room : [ Computer Lab 202]

Operating System : [Tuffix]

To be filled out by the Instructor:

GRADE:

COMMENTS:

# CS323 Documentation

## 1. Problem Statement

Write a syntax analyzer. You may use any top-down parser such as a TDP, a predictive table driven predictive parser.

- Rewrite the grammar on the page 116 of the textbook to remove any left recursion.
- Use the lexer() generated in the assignment 1 to get the tokens.
- The parser should print to an output file the tokens, lexemes and the production rules used; That is, first, write the token and lexeme found. Then, print out all productions rules used for analyzing this token.
- Error handling: if a syntax error occurs, the parser will generate a meaningful error message, an error type between syntax or token type. Then, parser continues for further analysis.

## 2. How to use the program

Usage :

- 1) In terminal, move to the directory of executable file and input file
- 2) Compile with "g++ TDP\_YKim.cpp -o TDP"
- 3) In command line "./TDP InputfileName.txt"
- 4) In the same folder, the program will make a new "output.txt" file

## 3. Design of the program

**Main role of each parts :**

Read and write file : Donghao Feng

Data Structures planning : Haojie Pan

Flow Design : Yong Kim

Table Design : Donghao Feng / Haojie Pan / Yong Kim

Program Test : Donghao Feng

## 4. Any Limitation

Cannot handle "if else" statement, "while", and "functions" for now.

## 5. Any shortcomings

None.