Spring 2018 CIS 335 Assignment 7

(Due May 4)

Project Description

In this project you are asked to write a small compiler for a very simple C-like language using the **Top-Down (Recursive Descent)** parsing method. The simplified grammar rules are given in the next page of this assignment. Your compiler reads a program from a file. The input file name is given from the command line thus:

./csucc sample.c

Your compiler csucc rends the file sample.c, checks its syntax, and generates the corresponding SIC/ XE assembly code into the file sample.asm.

To reduce your work, you may assume that there is at least a space character between consecutive tokens. Furthermore, you only need to implement what the simplified grammar specifies. That is, your program does not need to support explicit declaration, conditional and iteration statements, etc.

Submission

Each student needs to hand in a typed document describing how your compiler is implemented (including the data structures, modules, etc.) and the bugs during the implementation to the blackboard. The cover page should include your name and your photo ID.

You are to submit your program electronically as well in addition to the project description document. You may submit your file more than once if you have need to update your program but only the last submission will be considered.

Start early and good luck.

---------- Grammar Rules ---------

<stmt-list> := <stmt> { <stmt>}

<stmt> := id= <expr>;

<expr> ::= <term> { + <term> | - <term>}

<term> ::= <factor> { * <factor> | / <factor> }

<factor> := id | intnum | (<expr>)

---------- Sample Program ----------

X = 5;

Y = 4 * X;

Z = X * 2 - (Y + X) / 5;

---------- Corresponding SIC/XE Output ----------

```
LDA    #5
STA    X
LDA    #4
MUL    X
STA    y
LDA    X
HUL    #2
STA    T1
LDA    y
ADD    X
DIV    #5
STA    T2
LDA    T1
SUB    T2
STA    Z

X      RESW  1
Y      RESW  1
Z      RESW  1
Tl     RESW  1
T2     RESW  1
```