# COMP 215 C Programming III/Project

## Skills Assessed:

Construct a recursive descent parser for a context-free language

## Task:

Develop a recursive-descent parser to evaluate arithmetic expressions defined by the following context-free grammar described in EBNF:

command *→* expr ‘\n’

expr *→* term { + term | - term }

term *→* power { *∗* power | / power | % power }

power → factor [ ^ power ]

factor *→* [-] factor1

factor1 *→* ( expr ) | NUMBER

NUMBER = [0 - 9]+

The parser needs to be developed without using libraries such as flex and bison.

The following *enumeration* for TokenType is suggested:

typedef enum {PLUS, MINUS, DIVIDE, MULT, REMAINDER, POWER, LPAREN, RPAREN, NUMBER, ERROR, EOL} TokenType;

Use the following struct definition for Token:

struct Token {

TokenType type; int value;

}

You may structure your program as consisting of the following functions:

* + int expr(void);
  + int term(void);
  + int power(void);
  + int factor(void);
  + int factor1(void);
  + void error(void);
  + void parse(void);
  + void match(TokenType tkType);
  + struct Token getToken(void);
  + int main();

Your executable should run from the command line. Two sample executions are shown below:

Sample Execution 1:

Input: 5 + 6 *∗* 2 *∧* 2 *∧* 2

Output:

5 NUMBER

+ PLUS

6 NUMBER

*∗* MULT

2 NUMBER

*∧* POWER

2 NUMBER

*∧* POWER

2 NUMBER Result: 101

Sample Execution 2:

Input: 19 % 5 *∗* 10 / ( 2 *∧* 2 ) Output:

19 NUMBER

% REMAINDER

5 NUMBER

*∗* MULT

10 NUMBER

/ DIVIDE

( LPAREN

2 NUMBER

*∧* POWER

2 NUMBER

) RPAREN Result: 10

## Deliverables:

Your program must compile and execute to be graded according to the rubric. Prepare a short paper that describes your solution to this task.

Compress all program files and the short paper into a zip file called “your\_fullname\_c3.zip" and submit it via My Franklin.

Pay close attention to the due date.

## Terms:

You are required to write your own program. This is the only way to learn. Copying or referencing other’s work including resources online is strictly prohibited and will be punished as cheating. If you need help, ask the instructor.

This is NOT a group assignment. You may discuss with other students but you have to work your own solution. Submissions with unreasonably high similarity will be treated as cheating and all parties involved will be charged per the University policy.