

# Homework #6

## Question 1 (6 pt.)

The following augmented grammar with numbered production rules accepts function declarations such as:

```
void x();  
int f(int x);  
float f2(int a, int b, float c);
```

- (1)  $F' \rightarrow F$
- (2)  $F \rightarrow \text{ty id } ( A )$
- (3)  $A \rightarrow \epsilon$
- (4)  $A \rightarrow \text{ty id } B$
- (5)  $B \rightarrow \epsilon$
- (6)  $B \rightarrow , \text{ ty id } B$

- a) Provide the FIRST and FOLLOW sets for each nonterminal in the grammar.
- b) Calculate the LR(0) automaton for the grammar.
- c) Obtain an SLR parse table for the grammar.
- d) Is this an SLR grammar? Justify your answer.

Upload written answers and diagrams for these questions in a PDF file named `hw6.pdf`.

## Question 2 (4 pt.)

Write a parser based on Flex and Bison for the grammar in the previous question. Upload the scanner in a file named `scanner.l`, and your parser in a file named `parser.y`. Your program should take a text file name as a command-line argument, and parse its content, providing the answer to whether the input file contains a sentence accepted by the grammar or not. Your program should compile correctly on the *fusion1.ece.neu.edu* machine by running the following sequence of commands:

```
$ bison -oparser.c parser.y -d -v  
$ flex -osscanner.c scanner.l  
$ g++ parser.c scanner.c -o parser
```