

## Project 2

### Parser

The goal of our project this semester is to develop a prototype C compiler. This is done by compiling C programs into processors such as x86, MIPS, ARM, or pseudo assemblies. In this class we will use **Java assembly code** (Jasmin, <http://jasmin.sourceforge.net/>) as the target codes. **The project is divided into several parts including language definition, lexical analyzer, C-grammar, symbol table handlings, parser, and code generation.** In the project 2, you will need to decide the set of language features you want to support in your compiler, and write the **syntax analyzer**.

Instead of generating codes, you will only need to output grammar rules in the code generator part to test whether the grammar is correctly passed. An example code may look like:

```
program: VOID ID '(' ')' '{' declarations stmts '}'
    { if (TRACEON)
        System.out.println("program: VOID ID ( ) { declarations stmts }"); }

declarations: type ID ';' declarations
    { if (TRACEON)
        System.out.println("declarations: type ID ; declarations"); }
...

```

You can find a demo C grammar in this link:

<http://wwwantlr.org/download/examples-v3.tar.gz>.

In your hand-in report, you need to have the followings:

- **To define the subset of the language which you want to choose from C.**
- Give a set of testing programs that can illustrate the features of your subset of the language. (at least 3 test programs)
- Use the “**ANTLR**” to help you develop the parser.
- You can use **Java** or **other languages** to write your parser. (**Java is recommended**)
- Please ensure your program can be executed under the **mcore8** or **linux.cs.ccu.edu.tw** workstation.

Please turn in the following:

- A file describes your language which is a subset of C language. (MS-WORD file)
- The source codes:
  - ANTLR grammar file, `myparser.g`.
  - A program to call your parser, `testParser.java`.
  - Testing programs. (at least 3 programs)
- A readme file (pure text file) describes how to compile and execute your parser analyzer.
- A “Makefile”.

Due Date: May 10 (Friday), 24:00pm, 2019