# CSCI 4160 Project3

Due: see class calendar

**Goal:**

This assignment serves several purposes:

- to be familiar with Bison;
- to create the context free grammar for Tiger language.

**Description:**

This is a project that builds on the previous project. In this project you are required to use Bison to build a parser for the Tiger language. More specifically, write the context free grammar for the Tiger language. Please read Tiger language manual carefully before you start. The manual can be found in the class repository\Document\Tiger folder.

**What to do in this project?**

Come up with context free grammar for Tiger language, and use Bison to generate the parser for it. Your grammar should have as few shift-reduce conflicts as possible, and no reduce-reduce conflicts allowed. When you compile your parser file, Bison will output information about the number of shift-reduce conflicts as well as the number of reduce-reduce conflicts. If your grammar has a reduce-reduce conflict, please talk to me to see how to get rid of it.

A draft of Tiger CFG can be found in the brief Tiger manual located at "class repository\Document\Tiger\BriefManual.pdf". However, some changes are needed to make it work.

In the tiger.yy file, the precedence level and association of operators should be specified based on the manual.

Extra rules should be provided to recover from errors. You grammar should generate the same output on the sample file tiger0.tig as the one provided by the instructor.

**Set up environment:**

I strongly suggest you to use the sample Visual Studio solution provided by me in the class repository.

1. There is no more tiger.ll file since the instructor provides lex.yy.cc file.
2. Put your grammars in tiger.yy file, which is the only file you need to work on.
3. To compile your project,
    a. Compile Tiger.yy.
    b. Build the MainDriver project.
4. There are two ways to debug your grammar,
    a. In the sample solution provided by the instructor, there is a file "tiger.output" under "FlexBison Tools"\Resource Files within Visual Studio environment. This file is generated every time tiger.yy is parsed. It contains all state information of your grammars.
    b. Reset yydebug to 1 at line 29 at main.cpp file. This will generate debug information on terminal when compiling a tiger file. However, the debug information is probably hard to understand.

**Instructor provided files in the class repository**

The following files are provided by the instructor:

- Project3\BisonProject: This folder contains all source files and required visual studio files
    - Skeleton source files provided in the sample project are listed below:

        - lex.yy.cc: provided by instructor
        - tiger.yy: a skeleton file for tiger CFG.
        - tiger.tab.hh & tiger.tab.cc: generated by Bison when compiling tiger.yy.
        - tiger.output: debug file for CFG
        - ErrorMsg.h: contains the definition of error handler
        - main.cpp: the driver
        - tiger0.tig and tiger1.tig: test cases of Tiger language
- Description3.pdf: this file
- Rubric3.doc: the rubric used to grade this assignment.
- Tiger0.txt and tiger1.txt: expected output for tiger0.tig and tiger1.cl, respectively.

**How to submit**

1. Once you have finished, commit the project to the local repository and push it to the remote repository.
    - **Any commit of the project after the deadline is considered as cheating. If this happens, the latest version before the deadline will be graded, and you may receive up to 50 points deduction.**
2. You can check your overall grade by pull rubric3.doc from the master repository after the notice from the instructor.