Lab 4

COEN 175 Compilers

Overview For Lab 4

Goal

- Create a symbol table

Submission

- Submit a tarball of your cpps and make files in folder called phase3
- Due Date: Sunday February 6th

Main Objectives for the entire phase

* You will be given a working solution for phase 2
- Modify your parser
- Write a checker
- Make Symbol, Scope, and Type Classes

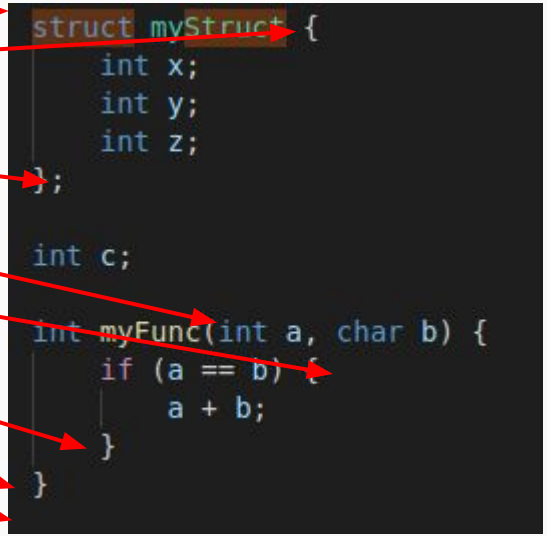
Goals for this week

1. Scope checking in parser.cpp
2. Create openScope() and closeScope() in checker.cpp/.h
3. Modify parser.cpp to pass information between functions
4. Write identifier() and number() functions that returns the identifier name and number respectively after matching
5. Make your Type class (from information provided in class)
6. Create the remainder of checker functions

1. Printing scopes

cout in parser for every time scopes are opened or closed

Open Scope
Open Scope
Close Scope
Open Scope
Open Scope
Close Scope
Close Scope
Close Scope



2. Writing checker.cpp/.h

- In lab:
  - openScope()
  - closeScope()
- cout open/close scope
- Before next lab (put in appropriate place in parser.cpp):
  - declareFunc()
  - defineFunc()
  - declareVariable()
  - checkIfStructure()
  - checkID()
- cout “name: type” for declare/define functions
- remove the cout from part 1

3. Modifying parser

- Return value from specifier()
- Pass in value into declarator()
- Return value from pointers()
- Given in class

4. identifier() and number()

- Code provided in class
- Matches ID and number respectively
- Returns appropriate value

5. Making your Type Class

- Should store the following
  - What the specifier type is (array, callback, error, function, or scalar)
  - How much indirection there is
  - If an array, note the length
  - If a function, parameter information
- Overload the == and != operators for equality checking
- Overload the ostream operator <<
- Print out in declare functions using the overloaded operator to test
- Type class was written in lecture

Tips

- A lot of code is/will be provided in class
- You should include boolean functions to check for each type
- Parameter information can be a list of Types
- identifier() should replace all instances of match(ID)
- number() will be used for declaring arrays
- Don’t forget to change all instances of your modified functions in parser to include a parameter or save the return value
- Example .out files are in /scratch/coen175/phase3_1/
- **READ THE SEMANTIC RULES CAREFULLY**