COEN 175

Phase 4 - Week 2

TAs

- Chris Desiniotis: cdesiniotis@scu.edu
  - Office Hours: Friday 12 - 2 pm
- Antonio Gigliotti: agigliotti@scu.edu
  - Office Hours: Thursday 11 - 1 pm

Extra Help/Tutoring

- Tau Beta Pi Tutoring
- Link to Tutoring schedule
  - https://sites.google.com/scu.edu/scutaubetapi/tutoring?authuser=1&pli=1

Phase 4 - Type Checking

Goal for this week:

- Finish the rest of checker (statements, function calls)

1. Statements
2. Function Calls

- Due February 21th 11:59PM

1. Statements

LINE 1: while (*expression*
LINE 2:)

- Test expressions (e.g. while, for, if)
  - Call your checker function right after the test expression is parsed
  - For example, for while loops, call the checker function before matching ‘)’
  - Why? If there is an error with the above test expression, we want to report the error on line 1, not line 2! Remember, lexer is responsible for incrementing the line number
- Return
  - Return type is always a scalar
  - How do we track the return type of a function? Can we use a global variable for this?
    - Remember: Simple C does not support nested function definitions
- Assignment
  - Call your checker function prior to matching ‘;’ (for same reasons as above)
  - Which lvalue matters for assignment? Make sure to check the lvalue we care about

2. Function Calls

- Gather arguments for function call in primaryExpression() (similar to how we handle parameters()) and make sure to pass these arguments to your checker function
- Suggested structure for checker function
    - Check if passed in type is a function
        - Return and report error if **not**
    - Check if parameters is nullptr
        - Return new Type if yes (valid function call)
    - Compare size of parameters (function definition) to size of arguments (function call)
        - Return and report error if **they differ**
    - Check to make sure each parameter is compatible with each argument
        - Return and report error if any are **not**
    - If you have gotten this far return new Type (valid function call)

Common Errors

- Misunderstanding ‘pointer to void’ vs ‘pointer to T’
- Not returning the right Type from checker
    - Make sure you write test cases which test this!
    - E.g. a || **b && c**
        - The return type of the ‘&&’ expression is used in the ‘||’ expression
- Check all combinations for operators