
Homework 1: Implementation

Dustin Graves

CS342

February 8, 2007

Tool Selection

- Modified version of lcc: A retargetable compiler for ANSI C
 - Very simple compiler described in the book "A Retargetable C Compiler: Design and Implementation"
 - Lex and Yacc based
 - Available from <http://www.cs.princeton.edu/software/lcc/>
- lcc provides a set of compiler components that are controlled by a “driver”
 - Available: pre-processor, compiler, code-generator, driver
 - Used: pre-processor, compiler
 - Modified: compiler
 - Input to compiler (rcc) must be output from pre-processor (cpp)

Modification

- Symbol table management code:
 - Added “initialized” field to symbol record
 - Initialize “initialized” field to FALSE at time of symbol insertion
- Lexical analysis code
 - Check “initialized” field of symbols encountered while evaluating an assignment expression
- Variable declaration code
 - If a variable is initialized at declaration, a global value indicating evaluation of assignment is expression is set
 - At completion of assignment evaluation, a message is generated if tokenizer indicates use of uninitialized symbol
- Expression evaluation code
 - If an assignment expression is detected, a global value indicating evaluation of assignment is expression is set
 - At completion of assignment evaluation, a message is generated if tokenizer indicates use of uninitialized symbol
 - When encountering an uninitialized variable during expression evaluation, a message is generated with line number of occurrence and line number of uninitialized variable declaration

Operation

- Basic operation
 - Execute pre-processor to generate input for compiler
 - `cpp file.c > file.c.i`
 - Execute compiler to check for assignment of uninitialized variables
 - `rcc -target=null file.c.i`
 - Setting target to null suppresses assembly code generation
- Advanced operation
 - A GUI was designed to make assignment of uninitialized variable attestation process easier to perform and understand