CSE 341: Compilers and Interpreters

Project Submission

Mike Berry (*berrym@seas.upenn.edu*)

1 High level description

The project consists of a compiler (*"mcc"*) that compiles a subset of the C language into SPARC Assembly language.

The compiler consist of three distinct phases:

1. lexical analysis
2. parsing
3. code generation

Type checking and symbol management are done throughout the three stages.

2 The source code

2.1 Management

The source code is managed by *RCS* and compiled using the *make* utility. The source code is divided up somewhat along the lines of the major functions described in section 1:

- globals.h – structures, global variabes, and function prototypes
- mcc.l – flex lexer
- mcc.y – bison parser
- symbol.c – symbol table management
- tree.c – syntax tree management, type checking
- register.c – code generation, register allocation

2.2 Compiling

Using the make utility, the code compiles into an executable called mcc.

```
$ make
bison -v -d mcc.y
gcc -g  symbol.c -c
gcc -g  tree.c -c
/usr/local/bin/flex mcc.l
gcc -g  lex.yy.c -c
gcc -g  mcc.tab.c -c
gcc -g  register.c -c
gcc -g  -o mcc symbol.o lex.yy.o mcc.tab.o tree.o register.o -lfl -ly
```

3 What works; what doesn't work

The primary weakness of *mcc* is inefficient code generation. Registers are flushed and loaded to and from the stack often; global registers are not used at all; local registers are not used often enough.

Also, global variables were not implemented correctly and dropped. I have also noticed that integer → float raising does not occur in some instances; however, it does occur correctly in all the sample programs. I was unable to isolate the source of the problem.

4 Submission

There are three "useful" programs which demonstrate commonly known non-trvial algorithms, and one "useless" program which demonstrates various control structures and expressions.

With each program is the mcc source, assembly output from mcc, and compile/ runtime shell logs.

1. fquick – quick sort
2. ssort – shell sort
3. factorial – factorial

4. sample – “useless”

Also attached is the full source listing.

5 Acknowledgements

I’d like to thank my roommates for putting up with me talking to myself while wearing a bicycle helmet while doing this project.