CMSC 430 Programming Exercise
Byte Code Optimizer
Due Date: Wed, Dec 11th, 11:59pm.

In this project you will add some optimizations to your C- code generator. Your tasks are to implement two classes of optimizations.

Peephole optimizations (during AST and bytecode generation):

- evaluate constant arithmetic expressions (e.g., 1+2 → 3)
- fold constant relational expressions (e.g., 1 == 1 → true)
- simplify boolean expressions (e.g., true && x → x)
- apply algebraic simplification (e.g., 0+x → x)
- simplify IF statements (e.g., IF (true) stmt1 ELSE stmt2 → stmt1)
- simplify WHILE loops (e.g., WHILE (true) stmts → L stmts GOTO L)

Global optimizations: (after bytecode generation):

1. build basic blocks
2. construct control flow graph
3. compute variable live ranges
4. eliminate dead code
5. assign local variable slots

Getting Started

There are additions and changes to the ClassFile.java file you should incorporate into your code from project 4. All your code changes for this project should be in mycc.cup and ClassFile.java. You can get a copy of the go scripts, test files, and new copy of the skeleton parser by typing:

    cp -r ~chtseng/proj5 ~/proj5

Submission Instructions

You can turn in your assignment using the submit program. To use submit, add the following line to your .login file.

    alias submit ~chtseng/bin/submit

To submit, go to your directory containing the code and type:

    submit 5 mycc.lex mycc.cup *.java

The submit program will accept multiple submissions up to the submission deadline, overwriting previous submissions. Feel free to submit your project as many times as you desire before the deadline. The late submission policy is: 20% penalty for first 24 hours, 10% each additional day. Maximum 5 days late.