

15-745 Advanced Optimizing Compilers: Task 2

Due 2/23/2006 5pm

February 10, 2006

1 Task Description

In this assignment, you will implement at least one classical loop optimization.

2 Task Specifics

Please implement loop-invariant code motion (hoisting) as described in class. Note that this optimization uses reaching definitions and liveness, which you should have already implemented in Task 1. Implement the optimizations on top of your existing compiler from Task 1. Implementing this optimization will be sufficient to earn you full credit on this assignment.

Additionally, we will be running a small contest during this assignment. There are eight “challenge tests” (listed below) from the `big` subdirectory in the test suite that we will use to evaluate your compiler. Your performance relative to other groups in the class will determine in what order you get to select time slots for your project proposal presentation. You may implement any other loop-based optimizations you want (Induction Variable Elimination, Bounds Check Elimination, Null-Check Elimination, etc). We will be posting a running leaderboard, as well as instructions for submitting your times, during the second week of the assignment. Details will be announced via the website and RSS feed.

- `13_test2.13`: prime number generation. simple arrays
- `13_test3.13`: cellular automata. good array test
- `13_test8.13`: designed specifically for bounds-check elimination
- `13_test10.13`: simple loops that access an array
- `13_test11.13`: pi to 1000 digits. simple array access, plus CSE
- `13_test13.13`: n-queens. array and struct element access
- `13_test14.13`: n^2 -queens. much more demanding
- `13_test20.13`: busy arrays

Use the algorithm descriptions given in class, or in Muchnick. You may work in groups of one or two (we suggest two). Once you have completed the task, please turn in (by email to `mderosa@cs`) the a tarball of your modified source code and the `results.txt` file from a full test suite run (using the original Task 0 test suite).