

Compiler Construction WS09/10

Exercise Sheet 8

Please hand in the solutions to the theoretical exercises until the beginning of the lecture next Wednesday 2010-01-06, 10:00. Please write the number of your tutorial group or the name of your tutor on the first sheet of your solution. Solutions submitted later will not be accepted.

Exercise 8.1: I'll never finish this project ... (4 bonus points)

Formally prove the following claim. You cannot construct an optimal compiler which translates every given input program to a program with the minimal number of instructions.

Exercise 8.2: Intermediate Representation & Program Optimization (Project)

In the next phase of your project you are to transform the AST into the intermediate program representation Firm. For this you have to construct the type hierarchy in Firm as well as program graphs for every method. There will be an introduction to Firm in the tutorials next week (2009-12-17 and 2009-12-18).

Furthermore, you shall implement several optimizations:

1. constant propagation (as shown in the lecture),
2. common subexpression elimination (as shown in the lecture),
3. control-flow simplification (will be explained in the tutorials).

Of course you may add additional program optimizations to your compiler.

Send your solution to the practical exercise to jherter@cs.uni-sb.de until 2010-01-13 10:00. Please send just one e-mail per project group.