

Compiler Construction WS09/10

Exercise Sheet 7

Please hand in the solutions to the theoretical exercises until the beginning of the lecture next Wednesday 2009-12-16, 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 7.1: Static Program Analysis (Points: 4+4)

In this exercise, you are to design two static program analyses following the approach presented in the lecture. I.e. define the domain of your analysis (explicitly define the lattice the analysis is working on), give transfer and merge functions, and state how the system of equations for a fixpoint iteration is built up.

Assume the following program labels to exist:

- *NOP*, the no-operation, doing nothing
- *CHECK*(*expr*), this label checks a condition
- *ASSIGN*(*x*, *expr*), assignment to variable *x*
- *LOAD*(*memaddr*, *x*), loading the contents of memory address *memaddr* into variable *x*
- *STORE*(*memaddr*, *expr*) writing the value of *expr* to memory address *memaddr*

Define the following analyses:

1. Parameter Analysis

Design an analysis that identifies the variables of a program that are parameters. Use the following definition for parameter: Parameters are variables that are used before set on at least one path from the start point of a program to the end point.

2. Sign Analysis

Design an analysis that determines the sign of the values of program variables.

Exercise 7.2: Dominance (Points: 2)

Prove that for functions, where all statements are reachable from the start, the direct dominator relation forms a tree.