Prof. Dr. Reinhard Wilhelm
Prof. Dr. Sebastian Hack
Dipl.-Inform. Daniel Grund
Michael Jacobs, M.Sc.

Compiler Construction WS11/12

Exercise Sheet 9

Please hand in the solutions to the theoretical exercises until the beginning of the second Friday lecture after the Christmas break, 2012-01-20, 12:00. Please write the number of your tutorial group or the name of your tutor on the first sheet of your solution.

Exercise 9.1 Partial Orders and Lattices (Points: 2+2+4)

Prove the following statements.

- Let $(P, \sqsubseteq)$ be a partially ordered set. There can at most be one element $\bot \in P$ such that $\bot \sqsubseteq x$ for all $x \in P$.
- Let $(P, \sqsubseteq)$ be a partially ordered set. Consider the definitions of upper and lower bounds (slide 5 in the slide set about lattices) with respect to $\sqsubseteq$. Then it holds that $\emptyset^u = \emptyset^\ell = P$.
- $(L1)$ and $(L1)^D$ (slide 8 in the slide set about lattices) hold.

Exercise 9.2 Reaching Definitions (Bonus-Points: 2+2+3+3)

Consider the following program S of the While-language.

```
[y := 10]^1;
[x := x + y]^2;
while [y > 0]^3 do (
  [y := y - 1]^4;
  [x := x + 1]^5
);
[y := x]^6
```

- Draw the control flow graph of program S.
- Provide the equation system for the reaching definitions analysis based on program S.
- Determine the result of the analysis on program S by doing a fixed point iteration on the equation system.
- Give an example for a definition of a variable, that reaches a program point according to the analysis result, but that would never reach that program point in a real run of the program. Why does this imprecision arise?

Exercise 9.3 Jingle Else (Bonus-Points: 2+8+2)

Consider the context-free grammar $G = (\{S, EXP, COND\}, \{if, then, else, id, !, <, >, =\}, P, S)$ with productions P defined as follows:

$$
\begin{array}{rcl}
S & \to & EXP \\
EXP & \to & if\ COND\ then\ EXP \\
EXP & \to & if\ COND\ then\ EXP\ else\ EXP \\
EXP & \to & id \\
COND & \to & !\ COND \\
COND & \to & id\ <\ id \\
COND & \to & id\ >\ id \\
COND & \to & id\ =\ id
\end{array}
$$

- Show that the grammar G is ambiguous by providing an example word from $L(G)$ that exhibits this ambiguity. Justify that the grammar G is really ambiguous with respect to that word.
- Provide the canonical $LR(1)$ finite state machine for G and point out which $LR(1)$ conflicts arise from the ambiguity of the grammar.
- This ambiguity problem is well known. Many language specifications solve it by informally specifying that each *else* is bound to the closest preceding *if* that is not yet bound with an *else*. Provide a straight forward manipulation of the lookahead sets of your $LR(1)$ finite state machine such that the PDA controlled by the resulting finite state machine is conflict free and results in the same parse behavior as the informal specification just given.