Software-Modellierung und Verifikation
Informatik 2
Prof. J.-P. Katoen
RWTH Aachen

Priv.-Doz. T. Noll noll@cs.rwth-aachen.de
C. Jansen christina.jansen@cs.rwth-aachen.de

3. Exercise sheet *Semantics and Verification of Software SoSe2010*

Due to Monday, 10th May 2010, *before* the exercise course begins.

Exercise 3.1: **(3 points)**

Show that the statement $c \in Cmd$ given by

$$y := 1; \textsf{while}\ \neg(x = 1)\ \textsf{do}\ (y := y * x;\ x := x - 1)$$

computes the factorial, i.e., that its operational semantics satisfies the following condition:

$$\mathfrak{O}[\![c]\!]\sigma(y) = (\sigma(x))!$$

for every $\sigma \in \Sigma$ with $\sigma(x) \geq 1$.

Exercise 3.2: **(2 points)**

In the lecture some concepts were introduced by means of inductive definitions, for example the syntax of WHILE-programs, the set of free variables in arithmetic expressions or recently the denotational semantics. Give an inductive definition of the length of commands. (*Hint:* You may assume, that keywords, numbers etc. have a standardized length of 1.)

Exercise 3.3: **(2 points)**

To summarize the operational semantics part of the lecture, answer the following questions.

(a) What is the effect of a statement execution in operational semantics? In which way is it integrated into the semantics?

(b) Sketch the sticking points of the induction principle capable of proving propositions about the operational execution relation. What happens, if we decide to apply induction on the structure of the syntactic expression?

(c) Give the definition of the operational functional. What is the result of $\mathfrak{O}[\![c]\!]$ and $\mathfrak{O}[\![c]\!]\sigma$, $c \in Cmd$?

Exercise 3.4: **(1+2 points)**

We define a (total) function as a relation $f \subseteq X \times Y$, where for all $x \in X$, there exists exactly one $y \in Y$ such that $f(x) = y$.

- Formulate the conditions defining a function in a more precise, mathematical way.
- Show that the denotation for arithmetic expressions is indeed a (total) function.