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

Priv.-Doz. T. Noll noll@cs.rwth-aachen.de
D. Willems willems@cs.rwth-aachen.de

4. Exercise sheet *Semantics and Verification of Software 2007*

Due to Wed., 9 May 2007, *before* the exercise course begins.

Exercise 4.1:

(a) Show that the least upper bound of a chain (Definition 5.7) is unique (if it exists).

(b) Give a subset of $\Sigma \dashrightarrow \Sigma$ which does not have an upper bound.

Exercise 4.2:

Which of the following functionals of type $(\Sigma \dashrightarrow \Sigma) \to (\Sigma \dashrightarrow \Sigma)$ are montonic with respect to the partial order $\sqsubseteq$ given by graph inclusion?

(a) $\Phi_1(f) = f$

(b) $\Phi_2(f) = \begin{cases} g_1 & \text{if } f = g_2 \\ g_2 & \text{otherwise} \end{cases}$ (where $g_1, g_2 : \Sigma \dashrightarrow \Sigma$ with $g_1 \neq g_2$)

(c) $\Phi_3(f)(\sigma) = \begin{cases} f(\sigma) & \text{if } \sigma(x) \neq 0 \\ \sigma & \text{otherwise} \end{cases}$

Exercise 4.3:

Investigate

$$\mathfrak{C}[\![z := 0; \textsf{while } y \leq x \textsf{ do } (z := z + 1; x := x - y)]\!]$$

in analogy to the factorial example 7.3.

Exercise 4.4:

(a) Define the denotational semantics of the **repeat** c **until** b construct.

(b) Using this semantics, show that the following semantic equivalence holds:

$$\textsf{repeat } c \textsf{ until } b \sim c; \textsf{while } \neg b \textsf{ do } c.$$

(**Hint:** The proof can be given by complete induction over the fixpoint iteration index n.)