

# Übungsblatt 7

**Ausgabe:** 07.01.2019

**Abgabe:** 17.01.2019, 12:00 Uhr

## Organisatorisches

Es gelten die Hinweise von Übungsblatt 5.

## Hausübung

### Aufgabe 7.1 (Quizfragen)

**6 P.**

Beantworten Sie folgende Fragen zum SWK-Stoff. Begründen Sie Ihre Antworten *in einem kurzen Satz*.

- (a) Muss eine Systemanforderung in jedem Fall eine Benutzerinteraktion beschreiben? 0,5 P.
- (b) Liegt ein Knoten bei der Methode der kritischen Pfade genau dann auf einem kritischen Pfad, wenn sein Puffer 0 ist? 0,5 P.
- (c) Sind alle Commitgraphen kreisfrei? 0,5 P.
- (d) Schließen sich Sharding und Load Balancing grundsätzlich aus? 0,5 P.
- (e) Lässt sich Continuos Deployment im Wasserfall-Modell realisieren? 0,5 P.
- (f) Sind UML-Objektdiagramme geeignet, um eine Anwendungsdomäne zu modellieren? 0,5 P.
- (g) Können zu einer Modellinstanz mehrere Modelle existieren? 0,5 P.
- (h) Gibt es Modellinstanzen, die Instanzen ihres Metametamodells sind? 0,5 P.
- (i) Ist die Semantik der Programmiersprache Haskell im Compiler realisiert? 0,5 P.
- (j) Existiert zu jedem deterministischen erweiterten endlichen Automaten ein deterministischer erweiterter endlicher Automat, der keine Zustandsvariablen benutzt? 0,5 P.
- (k) Gibt es immer eine eindeutige Lösung für die Reaching-Definitions-Analyse? 0,5 P.
- (l) Sind Unit Tests ein Beispiel für überapproximierende Analyse? 0,5 P.

### Aufgabe 7.2 (Strukturelle Operationelle Semantik)

8 P.

Nutzen Sie die SOS-Semantik, um folgendes Programm in erweiterter While-Syntax auszuwerten:

```
[i := 42]1;  
do  
  | for [j := 23]2 test [j > 0]3 update [j := j - 20]4 do  
  |   | [i := i + j]5  
while [i < 60]6
```

Dazu müssen Sie allerdings die Semantik erweitern, um die zusätzlichen Programmkonstrukte auswerten zu können.

- (a) Definieren Sie eine oder mehrere Auswertungsregeln, um **do...while...** korrekt auszuwerten. Das Statement soll sich wie ein **while...do...** verhalten, aber erst nach dem Block die Bedingung überprüfen. 1 P.
- (b) Geben Sie eine oder mehrere Auswertungsregeln an, um auch das Konstrukt **for...test...update...do** auszuwerten. Die einzelnen Komponenten des **for** verstehen sich wie in C oder Java. 1 P.

**Hinweis:** Benutzen Sie die alternative Darstellung von Übungsblatt 5.

- (c) Benutzen Sie die Auswertungsregeln der SOS und Ihre Erweiterungen, um das Programm auszuwerten. Die Auswertung von Rechenausdrücken müssen Sie nicht aufschlüsseln. Sie können im Interesse der Lesbarkeit Ausdrücke durch ihr Label ersetzen. 6 P.

**Beispiel:** Wir werten  $[x := 1]^1$ ; **if**  $[x = 1]^2$  **then**  $[x := 0]^3$  **else**  $[x := -x]^4$  aus:

$$\begin{array}{c} \langle 1, \emptyset \rangle \\ \xrightarrow{[\text{ASS}]} \{x \mapsto 1\} \end{array} \quad (1)$$

$$\begin{array}{c} \langle 1; \text{if } 2 \text{ then } 3 \text{ else } 4, \emptyset \rangle \\ \xrightarrow{[\text{COMP}_2], (1)} \langle \text{if } 2 \text{ then } 3 \text{ else } 4, \{x \mapsto 1\} \rangle \\ \xrightarrow{[\text{IF}_T]} \langle 3, \{x \mapsto 1\} \rangle \end{array}$$

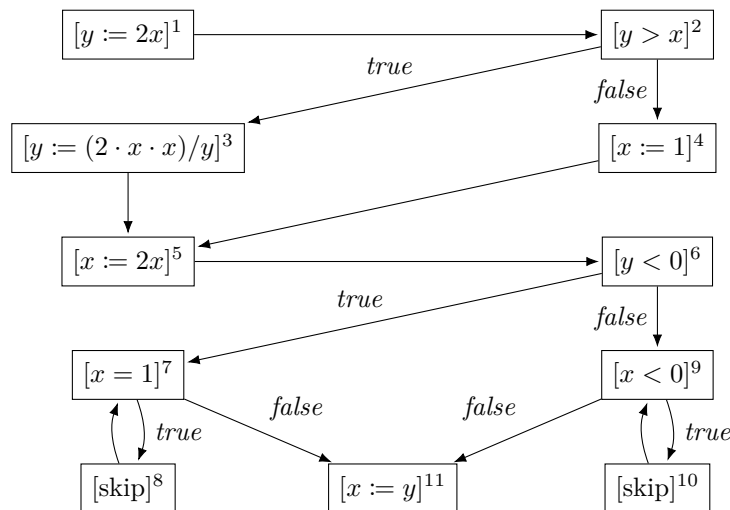
$$\xRightarrow{[ASS]} \{x \mapsto 0\}$$

Sie müssen also für jede Anwendung der Kompositionsregel eine Nebenrechnung für den ersten Teil ausführen<sup>1</sup>.

### Aufgabe 7.3 (Symbolische Ausführung)

6 P.

Sie wollen das folgende Programm auf Korrektheit untersuchen und benutzen dazu die Technik der symbolischen Ausführung, um Angaben über die Ergebnisse zu erhalten.



- (a) Führen Sie das Programm symbolisch aus. Geben Sie vor jedem Schritt an, welche Instruktion Sie „ausführen“ werden, welche symbolische Belegung existiert und welche Pfadbeschränkungen gelten. Wenn Sie in einem Schritt mit SAT oder UNSAT abbrechen, vermerken Sie dies. Geben sie im SAT-Fall ein Modell an. 4 P.

Sie müssen nicht die Baumdarstellung wählen, eine lesbare Tabelle genügt.

- (b) Überprüfen Sie anhand Ihrer Ergebnisse, ob in jedem Zustand die Invariante  $x|y \vee y|x$  ( $x$  teilt  $y$  oder  $y$  teilt  $x$ ;  $\exists d \in \mathbb{Z} : dx = y \vee dy = x$ ) gilt. Falls ja, erläutern Sie in eigenen Worten, wie die symbolische Ausführung zur Überprüfung dieser Invariante genutzt werden kann. Falls nein, nennen Sie ein Gegenbeispiel. 1 P.
- (c) Entscheiden Sie jetzt, ob das Programm in jedem Fall terminiert. Begründen Sie Ihre Antwort. 1 P.

<sup>1</sup>Eigentlich würde man hier einen Beweisbaum notieren. Aus Gründen der Lesbarkeit ist eine Nebenrechnung mit einer Anmerkung am Schritt-Pfeil aber für diese Übung ausreichend.