

## Semantik von Programmiersprachen – SS 2010

<http://pp.info.uni-karlsruhe.de/lehre/SS2010/semantik>

### Blatt 10: Fixpunkttheorie

Besprechung: 22.06.2010

#### 1. Welche der folgenden Aussagen sind richtig, welche falsch? (H)

- (a) Für  $f(\sigma) = \sigma[i \mapsto 1]$  und  $g(\sigma) = \sigma[i \mapsto 2]$  gilt  $f \sqsubseteq g$ .
- (b) Jede Teilmenge von  $\mathbb{R}_0^+ = \{x \in \mathbb{R} \mid x \geq 0\}$  hat bezüglich der normalen Ordnung  $<$  auf den reellen Zahlen  $\mathbb{R}$  ein kleinstes Element.
- (c) Jede Teilmenge einer total geordneten Menge ist eine Kette.
- (d) In einer ccpo  $(D, \sqsubseteq)$  hat jede Menge  $M \subseteq D$  eine obere Schranke.
- (e) Die Menge  $\mathfrak{P}^{fin}(\mathbb{N})$  der endlichen Teilmengen von  $\mathbb{N}$  ist mit der Teilmengenrelation  $\subseteq$  als Ordnung eine ccpo.
- (f) Jede ccpo  $(D, \sqsubseteq)$  hat ein kleinstes Element.
- (g) Das abgeschlossene Intervall  $[0, 1] \subseteq \mathbb{R}$  ist eine ccpo mit  $<$  als Ordnung.
- (h) IF  $(p, f, g)$  ist strikt in  $f$ .
- (i) Wenn  $f \circ g$  kettenstetig ist, dann sind auch  $f$  und  $g$  kettenstetig.

#### 2. Monotonie und Fixpunkte (H)

Finden Sie eine Halbordnung  $(D, \sqsubseteq)$  mit kleinstem Element  $\perp$  und eine monotone Funktion  $f :: D \Rightarrow D$ , die mehrere Fixpunkte besitzt, aber keinen kleinsten.

#### 3. repeat c until b-Schleife (Ü)

In einem früheren Aufgabe haben wir schon die operationale Semantik einer **repeat**-Schleife betrachtet. Com wird dazu um das Syntaxkonstrukt **repeat c until b** erweitert und die operationale Big-Step-Semantik durch die Regeln

$$\text{REPEATTT: } \frac{\langle c, \sigma \rangle \Downarrow \sigma' \quad \mathcal{B} \llbracket b \rrbracket \sigma' = \mathbf{tt}}{\langle \text{repeat } c \text{ until } b, \sigma \rangle \Downarrow \sigma'}$$

$$\text{REPEATFF: } \frac{\langle c, \sigma \rangle \Downarrow \sigma' \quad \mathcal{B} \llbracket b \rrbracket \sigma' = \mathbf{ff} \quad \langle \text{repeat } c \text{ until } b, \sigma' \rangle \Downarrow \sigma''}{\langle \text{repeat } c \text{ until } b, \sigma \rangle \Downarrow \sigma''}$$

- (a) Leiten Sie daraus die Rekursionsgleichung für  $\mathcal{D} \llbracket \text{repeat } c \text{ until } b \rrbracket$  her.
- (b) Erweitern Sie die Definition von  $\mathcal{D} \llbracket \_ \rrbracket$  um **repeat c until b**.
- (c) Prüfen Sie, ob die Semantik mit Ihrer Erweiterung weiterhin wohldefiniert und kompositional ist.
- (d) Zeigen oder widerlegen Sie:  $\mathcal{D} \llbracket \text{repeat } c \text{ until } b \rrbracket = \mathcal{D} \llbracket c; \text{ while } (\text{not } b) \text{ do } c \rrbracket$