

Übungsblatt 8

Aufgabe 1. Gegeben seien folgende Haskell-Funktionen:

```
three :: Int -> Int
three x = 3
```

```
times :: Int -> Int
times x = if x <= 0 then 0
          else x + times (x - 1)
```

```
inf :: Int -> Int
inf x = inf (x + 2)
```

- (a) In der Vorlesung haben wir `fact` definiert, was sich rekursiv aufruft. Danach haben wir `rfact` definiert, was stattdessen die rekursiv aufzurufende Funktion als Parameter erhält. Geben Sie analog zu `rfact` die Funktionen `rthree`, `rtimes` und `rinf` an.
- (b) Geben Sie jeweils eine geeignete Semantik für `rthree`, `rtimes` und `rinf` an.
- (c) Geben Sie jeweils $\llbracket \text{rthree} \rrbracket^i(\perp)$, $\llbracket \text{rtimes} \rrbracket^i(\perp)$ und $\llbracket \text{rinf} \rrbracket^i(\perp)$ für $i \in \mathbb{N}$ an.
- (d) Geben Sie alle Fixpunkte von $\llbracket \text{rthree} \rrbracket$, $\llbracket \text{rtimes} \rrbracket$ und $\llbracket \text{rinf} \rrbracket$ an. Was sind $\mu \llbracket \text{rthree} \rrbracket$, $\mu \llbracket \text{rtimes} \rrbracket$ und $\mu \llbracket \text{rinf} \rrbracket$?

Aufgabe 2. Sei (D, \sqsubseteq_D) eine CPO und $c: \mathbb{N} \rightarrow D$ eine Kette. Sei des Weiteren $c': \mathbb{N} \rightarrow D$ eine Kette mit $c'(i) = c(i + 1)$. Zeigen Sie, dass $\sqcup c = \sqcup c'$.

Aufgabe 3. Zeigen Sie, dass $\llbracket \text{rfact} \rrbracket(\text{fact}_i) = \text{fact}_{i+1}$ für alle $i \in \mathbb{N}$. Wie in der Vorlesung sind $\text{fact}_i: \mathbb{Z}_\perp \rightarrow \mathbb{Z}_\perp$ und $\llbracket \text{rfact} \rrbracket: (\mathbb{Z}_\perp \rightarrow \mathbb{Z}_\perp) \rightarrow (\mathbb{Z}_\perp \rightarrow \mathbb{Z}_\perp)$ definiert als $\text{fact}_0(x) = \perp$,

$$\text{fact}_{i+1}(x) = \begin{cases} \perp & \text{falls } x = \perp \text{ oder } x > i, \\ 1 & \text{falls } x < 0, \\ x! & \text{falls } 0 \leq x \leq i, \end{cases}$$
$$\llbracket \text{rfact} \rrbracket(f)(x) = \begin{cases} \perp & \text{falls } x = \perp, \\ 1 & \text{falls } x \leq 0, \\ x \cdot f(x - 1) & \text{falls } x > 0. \end{cases}$$