Exercise 3: Denotational and Operational Semantics

Concurrency Theory — Summer 2024

Dr. Stephan Mennicke

May 8th, 2024

TU Dresden, Knowledge-Based Systems Group

Exercise 2.4

Let g_1, g_2, and g_3 be defined as follows:

$$g_1\ s = \begin{cases} s & \text{if } s\,x \text{ is even} \\ \texttt{undef} & \text{otherwise} \end{cases}$$

$$g_2\ s = \begin{cases} s & \text{if } s\,x \text{ is a prime} \\ \texttt{undef} & \text{otherwise} \end{cases}$$

$$g_3\ s = s$$

1. Determine the ordering between these partial functions.
2. Determine a partial function g_4 such that $g_4 \sqsubseteq g_1$, $g_4 \sqsubseteq g_2$, and $g_4 \sqsubseteq g_3$ (i.e., g_4 is a lower bound of $\{g_1, g_2, g_3\}$).
3. Determine a partial function g_5 such that $g_1 \sqsubseteq g_5$, $g_2 \sqsubseteq g_5$, and $g_5 \sqsubseteq g_3$ but g_5 is distinct from g_1, g_2, and g_3.

Exercise 2.6

(a) Consider the ccpo $(\mathbb{N}, \subseteq)$. Determine which of the following functions in $\mathbb{N} \to \mathbb{N}$ are monotone:

$$
\begin{aligned}
f_1\, X &= \mathbb{N} \setminus X \\
f_2\, X &= X \cup \{27\} \\
f_3\, X &= X \cap \{7, 9, 13\} \\
f_4\, X &= \{n \in X \mid n \text{ is a prime}\} \\
f_5\, X &= \{2 \cdot n \mid n \in X\}
\end{aligned}
$$

(b) Which of the following functionals of

$$(\mathbf{State} \hookrightarrow \mathbf{State}) \to (\mathbf{State} \hookrightarrow \mathbf{State})$$

are monotone:

Exercise 2.6

$$F_0\, g = g$$

$$F_1\, g = \begin{cases} g_1 & \text{if } g = g_2 \\ g_2 & \text{otherwise} \end{cases}$$

$$(F_2\, g)s = \begin{cases} g\, s & \text{if}\, s\, x \neq 0 \\ s & \text{if}\, s\, x = 0 \end{cases}$$

(c) Show that F_2 is even continuous.

Exercise 3.1: A Case for Rule Induction

Lemma 1: The natural semantics is deterministic.

Proof: **Exercise** ■

Theorem 2: The semantic function of the natural semantics $\mathcal{S}_{\mathrm{ns}}[\![\cdot]\!] : \mathbf{Stm} \to (\mathbf{State} \hookrightarrow \mathbf{State})$ given by

$$\mathcal{S}_{\mathrm{ns}}[\![S]\!]\, s = \begin{cases} s' & \text{if } \langle S, s \rangle \Rightarrow s' \\ \mathtt{undef} & \text{otherwise} \end{cases}$$

exists (and is well-defined).

Exercise 3.2: An Equivalence Result

Theorem 3: The natural semantics and the direct style semantics coincide, that is

$$\mathcal{S}_{\mathrm{ds}}[\![S]\!] = \mathcal{S}_{\mathrm{ns}}[\![S]\!]$$

for all statements S of the While-language.

Proof: **Exercise** ■

Lemma 4: The structural operational semantics is deterministic.

$$\mathcal{S}_{\text{sos}}[\![S]\!]\, s = \begin{cases} s' & \text{if } \langle S, s\rangle \Rightarrow^* s' \\ \texttt{undef} & \text{otherwise} \end{cases}$$

Theorem 5: For all statements S, $\mathcal{S}_{\text{ns}}[\![S]\!] = \mathcal{S}_{\text{sos}}[\![S]\!]$.

Direct Consequence: All three semantics are equivalent.

The Direct Style Semantics in One Slide

- $\mathcal{S}_{\text{ds}}[\![x \texttt{ := } a]\!]\, s := s[x \mapsto \mathcal{A}[\![a]\!]\, s]$
- $\mathcal{S}_{\text{ds}}[\![\texttt{skip}]\!] := \text{id}$
- $\mathcal{S}_{\text{ds}}[\![S_1 \texttt{ ; } S_2]\!] := \mathcal{S}_{\text{ds}}[\![S_1]\!] \circ \mathcal{S}_{\text{ds}}[\![S_1]\!]$
- $\mathcal{S}_{\text{ds}}[\![\texttt{if } b \texttt{ then } S_1 \texttt{ else } S_2]\!] := \mathsf{cond}(\,\mathcal{B}[\![b]\!]\,, \mathcal{S}_{\text{ds}}[\![S_1]\!]\,, \mathcal{S}_{\text{ds}}[\![S_2]\!]\,)$
- $\mathcal{S}_{\text{ds}}[\![\texttt{while } b \texttt{ do } S]\!]\, s = \mathsf{FIX}\ F$

where $F\,g = \mathsf{cond}(\,\mathcal{B}[\![b]\!]\,, g \circ \mathcal{S}_{\text{ds}}[\![S]\!]\,, \text{id}\,)$

The Natural Semantics in One Slide

$$[\text{ass}_{\text{ns}}]\,\frac{}{\langle x := a,\, s\rangle \rightarrow s[x \mapsto \mathcal{A}[\![a]\!]\, s]} \quad [\text{skip}_{\text{ns}}]\,\frac{}{\langle \texttt{skip},\, s\rangle \rightarrow s} \quad [\text{seq}_{\text{ns}}]\,\frac{\langle S_1,\, s\rangle \rightarrow s'' \quad \langle S_2,\, s''\rangle \rightarrow s'}{\langle S_1 ; S_2,\, s\rangle \rightarrow s'}$$

$$[\text{if}^{\texttt{tt}}_{\text{ns}}]\,\frac{\langle S_1,\, s\rangle \rightarrow s'}{\langle \texttt{if}\ b\ \texttt{then}\ S_1\ \texttt{else}\ S_2,\, s\rangle \rightarrow s'} \quad \text{if } \mathcal{B}[\![b]\!]\, s = \texttt{tt}$$

$$[\text{if}^{\texttt{ff}}_{\text{ns}}]\,\frac{\langle S_2,\, s\rangle \rightarrow s'}{\langle \texttt{if}\ b\ \texttt{then}\ S_1\ \texttt{else}\ S_2,\, s\rangle \rightarrow s'} \quad \text{if } \mathcal{B}[\![b]\!]\, s = \texttt{ff}$$

$$[\text{while}^{\texttt{tt}}_{\text{ns}}]\,\frac{\langle S,\, s\rangle \rightarrow s' \quad \langle \texttt{while}\ b\ \texttt{do}\ S,\, s'\rangle \rightarrow s''}{\langle \texttt{while}\ b\ \texttt{do}\ S,\, s\rangle \rightarrow s''} \quad \text{if } \mathcal{B}[\![b]\!]\, s = \texttt{tt}$$

$$[\text{while}^{\texttt{ff}}_{\text{ns}}]\,\frac{}{\langle \texttt{while}\ b\ \texttt{do}\ S,\, s\rangle \rightarrow s''} \quad \text{if } \mathcal{B}[\![b]\!]\, s = \texttt{ff}$$

The Structural Operational Semantics (SOS) in One Slide

$$[\text{ass}_{\text{sos}}]\ \frac{}{\langle x \texttt{:=} a, s\rangle \Rightarrow s[x \mapsto \mathcal{A}[\![a]\!]\, s]}$$

$$[\text{skip}_{\text{sos}}]\ \frac{}{\langle \texttt{skip}, s\rangle \Rightarrow s}$$

$$[\text{seq}^1_{\text{sos}}]\ \frac{\langle S_1, s\rangle \Rightarrow \langle S_1', s'\rangle}{\langle S_1\texttt{;}S_2, s\rangle \Rightarrow \langle S_1'\texttt{;}S_2, s'\rangle}$$

$$[\text{seq}^2_{\text{sos}}]\ \frac{\langle S_1, s\rangle \Rightarrow s'}{\langle S_1\texttt{;}S_2, s\rangle \Rightarrow \langle S_2, s'\rangle}$$

$$[\text{if}^{\texttt{tt}}_{\text{sos}}]\ \frac{}{\langle \texttt{if}\ b\ \texttt{then}\ S_1\ \texttt{else}\ S_2, s\rangle \Rightarrow \langle S_1, s\rangle}\quad \text{if } \mathcal{B}[\![b]\!]\, s = \texttt{tt}$$

$$[\text{if}^{\texttt{ff}}_{\text{sos}}]\ \frac{}{\langle \texttt{if}\ b\ \texttt{then}\ S_1\ \texttt{else}\ S_2, s\rangle \Rightarrow \langle S_2, s\rangle}\quad \text{if } \mathcal{B}[\![b]\!]\, s = \texttt{ff}$$

$$[\text{while}_{\text{sos}}]\ \frac{}{\langle \texttt{while}\ b\ \texttt{do}\ S, s\rangle \Rightarrow \langle \texttt{if}\ b\ \texttt{then}\ S\texttt{;while}\ b\ \texttt{do}\ S\ \texttt{else}\ \texttt{skip}, s\rangle}$$