Harvard School of Engineering and Applied Sciences — CS 152: Programming Languages

IMP; Denotational Semantics
Section and Practice Problems

Week 4: Tue Feb 19–Fri Feb 22, 2019

1 IMP

Consider the small-step operational semantics for IMP given in Lecture 5. Let σ_0 be a store that maps all program variables to zero.

(a) Find a configuration $\langle c, \sigma' \rangle$ such that $\langle \textbf{if } 8 < 6 \textbf{ then } \mathsf{foo} := 2 \textbf{ else } \mathsf{bar} := 8, \sigma_0 \rangle \longrightarrow \langle c, \sigma' \rangle$ and give a derivation showing that $\langle \textbf{if } 8 < 6 \textbf{ then } \mathsf{foo} := 2 \textbf{ else } \mathsf{bar} := 8, \sigma_0 \rangle \longrightarrow \langle c, \sigma' \rangle$.

Answer:

$$\dfrac{\dfrac{}{\langle 8 < 6, \sigma_0 \rangle \longrightarrow \langle \textbf{false}, \sigma_0 \rangle}}{\langle \textbf{if } 8 < 6 \textbf{ then } \mathsf{foo} := 2 \textbf{ else } \mathsf{bar} := 8, \sigma_0 \rangle \longrightarrow \langle \textbf{if false then } \mathsf{foo} := 2 \textbf{ else } \mathsf{bar} := 8, \sigma_0 \rangle}$$

(b) What is the sequence of configurations that

$$\langle \mathsf{foo} := \mathsf{bar} + 3; \textbf{if } \mathsf{foo} < \mathsf{bar} \textbf{ then skip else } \mathsf{bar} := 1, \sigma_0 \rangle$$

steps to? (You don't need to show the derivations for each step, just show what configuration $\langle \mathsf{foo} := \mathsf{bar} + 3; \textbf{if } \mathsf{foo} < \mathsf{bar} \textbf{ then skip else } \mathsf{bar} := 1, \sigma_0 \rangle$ steps to in one step, then two steps, then three steps, and so on, until you reach a final configuration.)

Answer:

$$\begin{array}{ll}
& \langle \mathsf{foo} := \mathsf{bar} + 3; \textbf{if } \mathsf{foo} < \mathsf{bar} \textbf{ then skip else } \mathsf{bar} := 1, \sigma_0 \rangle \\
\longrightarrow & \langle \mathsf{foo} := 0 + 3; \textbf{if } \mathsf{foo} < \mathsf{bar} \textbf{ then skip else } \mathsf{bar} := 1, \sigma_0 \rangle \\
\longrightarrow & \langle \mathsf{foo} := 3; \textbf{if } \mathsf{foo} < \mathsf{bar} \textbf{ then skip else } \mathsf{bar} := 1, \sigma_0 \rangle \\
\longrightarrow & \langle \textbf{if } \mathsf{foo} < \mathsf{bar} \textbf{ then skip else } \mathsf{bar} := 1, \sigma_0[\mathsf{foo} \mapsto 3] \rangle \\
\longrightarrow & \langle \textbf{if } 3 < \mathsf{bar} \textbf{ then skip else } \mathsf{bar} := 1, \sigma_0[\mathsf{foo} \mapsto 3] \rangle \\
\longrightarrow & \langle \textbf{if } 3 < 0 \textbf{ then skip else } \mathsf{bar} := 1, \sigma_0[\mathsf{foo} \mapsto 3] \rangle \\
\longrightarrow & \langle \textbf{if false then skip else } \mathsf{bar} := 1, \sigma_0[\mathsf{foo} \mapsto 3] \rangle \\
\longrightarrow & \langle \mathsf{bar} := 1, \sigma_0[\mathsf{foo} \mapsto 3] \rangle \\
\longrightarrow & \langle \textbf{skip}, \sigma_0[\mathsf{foo} \mapsto 3, \mathsf{bar} \mapsto 1] \rangle
\end{array}$$

Now consider the large-step operational semantics for IMP given in Lecture 5. Let σ_0 be a store that maps all program variables to zero.

(c) Find a store σ' such that $\langle \textbf{while } \mathsf{foo} < 3 \textbf{ do } \mathsf{foo} := \mathsf{foo} + 2, \sigma_0 \rangle \Downarrow \sigma'$ and give a derivation showing that $\langle \textbf{while } \mathsf{foo} < 3 \textbf{ do } \mathsf{foo} := \mathsf{foo} + 2, \sigma_0 \rangle \Downarrow \sigma'$.

Answer:
In the following, let $\sigma_2 = \sigma_0[\mathsf{foo} \mapsto 2]$ *and* $\sigma_4 = \sigma_0[\mathsf{foo} \mapsto 4]$.

$$\dfrac{\dfrac{\dfrac{}{\langle \mathsf{foo}, \sigma_0\rangle \Downarrow 0} \quad \dfrac{}{\langle 3, \sigma_0\rangle \Downarrow 3}}{\langle \mathsf{foo} < 3, \sigma_0\rangle \Downarrow \textbf{true}} \quad \dfrac{\dfrac{\dfrac{}{\langle \mathsf{foo}, \sigma_0\rangle \Downarrow 0} \quad \dfrac{}{\langle 2, \sigma_0\rangle \Downarrow 2}}{\langle \mathsf{foo} + 2, \sigma_0\rangle \Downarrow 2}}{\langle \mathsf{foo} := \mathsf{foo} + 2, \sigma_0\rangle \Downarrow \sigma_2} \quad D_1}{\langle \textbf{while}\ \mathsf{foo} < 3\ \textbf{do}\ \mathsf{foo} := \mathsf{foo} + 2, \sigma_0\rangle \Downarrow \sigma_4}$$

where D_1 *is the following derivation*

$$\dfrac{\dfrac{\dfrac{}{\langle \mathsf{foo}, \sigma_2\rangle \Downarrow 2} \quad \dfrac{}{\langle 3, \sigma_2\rangle \Downarrow 3}}{\langle \mathsf{foo} < 3, \sigma_2\rangle \Downarrow \textbf{true}} \quad \dfrac{\dfrac{\dfrac{}{\langle \mathsf{foo}, \sigma_2\rangle \Downarrow 2} \quad \dfrac{}{\langle 2, \sigma_2\rangle \Downarrow 2}}{\langle \mathsf{foo} + 2, \sigma_2\rangle \Downarrow 4}}{\langle \mathsf{foo} := \mathsf{foo} + 2, \sigma_2\rangle \Downarrow \sigma_4} \quad D_2}{\langle \textbf{while}\ \mathsf{foo} < 3\ \textbf{do}\ \mathsf{foo} := \mathsf{foo} + 2, \sigma_2\rangle \Downarrow \sigma_4}$$

where D_2 *is the following derivation*

$$\dfrac{\dfrac{\dfrac{}{\langle \mathsf{foo}, \sigma_2\rangle \Downarrow 4} \quad \dfrac{}{\langle 3, \sigma_2\rangle \Downarrow 3}}{\langle \mathsf{foo} < 3, \sigma_2\rangle \Downarrow \textbf{false}}}{\langle \textbf{while}\ \mathsf{foo} < 3\ \textbf{do}\ \mathsf{foo} := \mathsf{foo} + 2, \sigma_4\rangle \Downarrow \sigma_4}$$

(d) Suppose we extend boolean expressions with negation.

$$b ::= \cdots \mid \textbf{not}\ b$$

(i) Give an inference rule or inference rules that show the (large step) evaluation of **not** b.

Answer:

$$\dfrac{\langle b, \sigma\rangle \Downarrow \textbf{false}}{\langle \textbf{not}\ b, \sigma\rangle \Downarrow \textbf{true}} \qquad \dfrac{\langle b, \sigma\rangle \Downarrow \textbf{true}}{\langle \textbf{not}\ b, \sigma\rangle \Downarrow \textbf{false}}$$

2 Denotational Semantics

(a) Give the denotational semantic for each of the following IMP programs. That is, express the meaning of each of the following programs as a function from stores to stores.

(i) $\mathsf{a} := \mathsf{b} + 5; \mathsf{a} := \mathsf{a} \times \mathsf{b}$

Answer: *We can implement the rules and simplify to obtain:*

$$\begin{aligned}\mathcal{C}[\![\mathsf{a} := \mathsf{b} + 5; \mathsf{a} := \mathsf{a} \times \mathsf{b}]\!]\sigma &= \sigma[\mathsf{a} \mapsto \sigma[\mathsf{a} \mapsto \sigma(\mathsf{b}) + 5](\mathsf{a}) \times \sigma(\mathsf{b})] \\ &= \sigma[\mathsf{a} \mapsto (\sigma(\mathsf{b}) + 5) \times \sigma(\mathsf{b})]\end{aligned}$$

(ii) $\textbf{if } \mathsf{foo} < 0 \textbf{ then } \mathsf{bar} := \mathsf{foo} \times \mathsf{foo} \textbf{ else } \mathsf{bar} := \mathsf{foo} \times \mathsf{foo} \times \mathsf{foo}$

Answer: *We take c to be the command above:*

$$\mathcal{C}[\![c]\!]\sigma = \begin{cases} \sigma[\mathsf{bar} \mapsto \sigma(\mathsf{foo}) \times \sigma(\mathsf{foo})] & \sigma(\mathsf{foo}) < 0 \\ \sigma[\mathsf{bar} \mapsto \sigma(\mathsf{foo}) \times \sigma(\mathsf{foo}) \times \sigma(\mathsf{foo})] & \textit{otherwise} \end{cases}$$

(iii) $\mathsf{bar} := \mathsf{foo} \times \mathsf{foo}; \textbf{if } \mathsf{foo} < 0 \textbf{ then skip else } \mathsf{bar} := \mathsf{bar} \times \mathsf{foo}$
(Hint: the answer to this question should be the same function as the answer to Question 2(a).ii above. You may have written the function down differently, but it should be the same mathematical function.)

Answer: *We have a very similar function to the above:*

$$\mathcal{C}[\![c]\!]\sigma = \begin{cases} \sigma[\mathsf{bar} \mapsto \sigma(\mathsf{foo}) \times \sigma(\mathsf{foo})] & \sigma(\mathsf{foo}) < 0 \\ \sigma[\mathsf{bar} \mapsto \sigma[\mathsf{bar} \mapsto \sigma(\mathsf{foo}) \times \sigma(\mathsf{foo})](\mathsf{bar}) \times \sigma(\mathsf{foo})] & \textit{otherwise} \end{cases}$$

(iv) $\mathsf{a} := 0; \mathsf{b} = 0; \textbf{while } \mathsf{a} < 3 \textbf{ do } \mathsf{b} := \mathsf{b} + \mathsf{c}$

Answer: *Note that the above term diverges. So $\mathcal{C}[\![\mathsf{a} := 0; \mathsf{b} = 0; \textbf{while } \mathsf{a} < 3 \textbf{ do } \mathsf{b} := \mathsf{b} + \mathsf{c}]\!]$ is the partial function with an empty domain.*

(b) Consider the following loop.

$$\textbf{while } \mathsf{foo} < 5 \textbf{ do } \mathsf{foo} := \mathsf{foo} + 1; \mathsf{bar} := \mathsf{bar} + 1$$

We will consider the denotational semantics of this loop.

(i) What is the denotational semantics of the loop guard $\mathsf{foo} < 5$? That is, what is the function $\mathcal{B}[\![\mathsf{foo} < 5]\!]$?

Answer:

$$\mathcal{B}[\![\mathsf{foo} < 5]\!] = \{(\sigma, \textbf{true}) \mid \sigma(\mathsf{foo}) < 5\} \cup \{(\sigma, \textbf{false}) \mid \sigma(\mathsf{foo}) \geq 5\}$$

Equivalently:

$$\mathcal{B}[\![\mathsf{foo} < 5]\!] = \begin{cases} \textbf{true} & \textit{if } \sigma(\mathsf{foo}) < 5 \\ \textbf{false} & \textit{if } \sigma(\mathsf{foo}) \geq 5 \end{cases}$$

(ii) What is the denotational semantics of the loop body $\mathsf{foo} := \mathsf{foo} + 1; \mathsf{bar} := \mathsf{bar} + 1$? That is, what is the function $\mathcal{C}[\![\mathsf{foo} := \mathsf{foo} + 1; \mathsf{bar} := \mathsf{bar} + 1]\!]$?

Answer: *After some simplification:*

$$\mathcal{C}[\![\mathsf{foo} := \mathsf{foo} + 1; \mathsf{bar} := \mathsf{bar} + 1]\!]\sigma = \sigma[\mathsf{foo} \mapsto \sigma(\mathsf{foo}) + 1, \mathsf{bar} \mapsto \sigma(\mathsf{bar}) + 1]$$

(iii) Recall that the semantics of the loop is the fixed point of the following higher-order function F. (This is from Section 1.2 of Lecture 6, where we have provided a specific loop guard b and loop body c for the higher-order function $F_{b,c}$).

$$\begin{aligned} F &: (\textbf{Store} \rightharpoonup \textbf{Store}) \to (\textbf{Store} \rightharpoonup \textbf{Store}) \\ F(f) &= \{(\sigma, \sigma) \mid (\sigma, \textbf{false}) \in \mathcal{B}[\![\mathsf{foo} < 5]\!]\} \;\cup \\ &\qquad \{(\sigma, \sigma') \mid (\sigma, \textbf{true}) \in \mathcal{B}[\![\mathsf{foo} < 5]\!] \wedge \\ &\qquad\qquad \exists \sigma''.\; ((\sigma, \sigma'') \in \mathcal{C}[\![\mathsf{foo} := \mathsf{foo} + 1; \mathsf{bar} := \mathsf{bar} + 1]\!] \wedge (\sigma'', \sigma') \in f)\} \end{aligned}$$

That is, the semantics of the loop are:

$$\begin{aligned} \mathcal{C}[\![\textbf{while}\ \mathsf{foo} < 5\ \textbf{do}\ \mathsf{foo} := \mathsf{foo} + 1; \mathsf{bar} := \mathsf{bar} + 1]\!] &= \bigcup_{i \geq 0} F^i(\emptyset) \\ &= \emptyset \cup F(\emptyset) \cup F(F(\emptyset)) \cup F(F(F(\emptyset))) \cup \ldots \\ &= \mathrm{fix}(F) \end{aligned}$$

Compute $F(\emptyset)$, $F(F(\emptyset))$, and $F(F(F(\emptyset)))$.
In general, what is the domain of the partial function $F^i(\emptyset)$? (Note that $F^i(\emptyset)$ is F applied to the empty set i times, e.g., $F^3(\emptyset)$ is $F(F(F(\emptyset)))$.)

Answer: *For reference as to how we arrive at the below, please see lecture notes.*

$$\begin{aligned} F(\emptyset) = &\{(\sigma, \sigma) \mid \sigma(\mathsf{foo}) \geq 5\} \\ F^2(\emptyset) = &\{(\sigma, \sigma) \mid \sigma(\mathsf{foo}) \geq 5\} \\ &\cup \{(\sigma, \sigma[\mathsf{foo} \mapsto \sigma(\mathsf{foo}) + 1, \mathsf{bar} \mapsto \sigma(\mathsf{bar}) + 1]) \mid \sigma(\mathsf{foo}) = 4\} \\ = &\{(\sigma, \sigma) \mid \sigma(\mathsf{foo}) \geq 5\} \\ &\cup \{(\sigma, \sigma[\mathsf{foo} \mapsto 5, \mathsf{bar} \mapsto \sigma(\mathsf{bar}) + 1]) \mid \sigma(\mathsf{foo}) = 4\} \\ F^3(\emptyset) = &\{(\sigma, \sigma) \mid \sigma(\mathsf{foo}) \geq 5\} \\ &\cup \{(\sigma, \sigma[\mathsf{foo} \mapsto \sigma(\mathsf{foo}) + 1, \mathsf{bar} \mapsto \sigma(\mathsf{bar}) + 1]) \mid \sigma(\mathsf{foo}) = 4\} \\ &\cup \{(\sigma, \sigma[\mathsf{foo} \mapsto \sigma(\mathsf{foo}) + 2, \mathsf{bar} \mapsto \sigma(\mathsf{bar}) + 2]) \mid \sigma(\mathsf{foo}) = 3\} \\ = &\{(\sigma, \sigma) \mid \sigma(\mathsf{foo}) \geq 5\} \\ &\cup \{(\sigma, \sigma[\mathsf{foo} \mapsto 5, \mathsf{bar} \mapsto \sigma(\mathsf{bar}) + 1]) \mid \sigma(\mathsf{foo}) = 4\} \\ &\cup \{(\sigma, \sigma[\mathsf{foo} \mapsto 5, \mathsf{bar} \mapsto \sigma(\mathsf{bar}) + 2]) \mid \sigma(\mathsf{foo}) = 3\} \end{aligned}$$

In general, we have:

$$
\begin{aligned}
F^i(\emptyset) = &\{(\sigma, \sigma) \mid \sigma(\mathsf{foo}) \geq 5\} \\
&\cup \{(\sigma, \sigma[\mathsf{foo} \mapsto 5, \mathsf{bar} \mapsto \sigma(\mathsf{bar}) + 1]) \mid \sigma(\mathsf{foo}) + 1 = 5\} \\
&\cup \{(\sigma, \sigma[\mathsf{foo} \mapsto 5, \mathsf{bar} \mapsto \sigma(\mathsf{bar}) + 2]) \mid \sigma(\mathsf{foo}) + 2 = 5\} \\
&\ldots \\
&\cup \{(\sigma, \sigma[\mathsf{foo} \mapsto 5, \mathsf{bar} \mapsto \sigma(\mathsf{bar}) + (i-1)]) \mid \sigma(\mathsf{foo}) + (i-1) = 5\}
\end{aligned}
$$

So we note that F^i is defined for all σ such that $\sigma(\mathsf{foo}) \geq i$.