1 Large-step operational semantics

In the last lecture we defined a semantics for our language of arithmetic expressions using a small-step evaluation relation $\rightarrow \subseteq \text{Config} \times \text{Config}$ (and its reflexive and transitive closure $\rightarrow^*$). In this lecture we will explore an alternative approach—large-step operational semantics—which yields the final result of evaluating an expression directly.

Defining a large-step semantics boils down to specifying a relation $\downarrow$ that captures the evaluation of an expression. The $\downarrow$ relation has the following type:

$$\downarrow \subseteq (\text{Store} \times \text{Exp}) \times (\text{Store} \times \text{Int}).$$

We write $\langle \sigma, e \rangle \downarrow \langle \sigma', n \rangle$ to indicate that $((\sigma, e), (\sigma', n)) \in \downarrow$. In other words, the expression $e$ with store $\sigma$ evaluates in one big step to the final store $\sigma'$ and integer $n$.

We define the relation $\downarrow$ inductively, using inference rules:

$$\frac{\langle \sigma, n \rangle \downarrow \langle \sigma, n \rangle}{\downarrow \text{ INT}} \quad \frac{n = \sigma(x)}{\langle \sigma \rangle \downarrow \langle \sigma, n \rangle \text{ VAR}}$$

$$\frac{\langle \sigma, e_1 \rangle \downarrow \langle \sigma', n_1 \rangle \quad \langle \sigma', e_2 \rangle \downarrow \langle \sigma'', n_2 \rangle \quad n = n_1 + n_2}{\langle \sigma, e_1 + e_2 \rangle \downarrow \langle \sigma'', n \rangle \text{ ADD}}$$

$$\frac{\langle \sigma, e_1 \rangle \downarrow \langle \sigma', n_1 \rangle \quad \langle \sigma', e_2 \rangle \downarrow \langle \sigma'', n_2 \rangle \quad n = n_1 \times n_2}{\langle \sigma, e_1 \times e_2 \rangle \downarrow \langle \sigma'', n \rangle \text{ MUL}}$$

$$\frac{\langle \sigma, e_1 \rangle \downarrow \langle \sigma', n_1 \rangle \quad \langle \sigma'[x \mapsto n_1], e_2 \rangle \downarrow \langle \sigma'', n_2 \rangle}{\langle \sigma, x := e_1 ; e_2 \rangle \downarrow \langle \sigma'', n_2 \rangle \text{ ASSGN}}$$

To illustrate the use of these rules, consider the following proof tree, which shows that evaluating $\langle \sigma, \text{foo} := 3 ; \text{foo} \times \text{bar} \rangle$ using a store $\sigma$ such that $\sigma(\text{bar}) = 7$ yields $\sigma' = \sigma[\text{foo} \mapsto 3]$ and 21 as a result:

$$\frac{\langle \sigma, 3 \rangle \downarrow \langle \sigma, 3 \rangle}{\downarrow \text{ INT}} \quad \frac{\langle \sigma', \text{foo} \rangle \downarrow \langle \sigma', 3 \rangle \text{ VAR}}{\langle \sigma', \text{foo \times bar} \rangle \downarrow \langle \sigma', 21 \rangle \text{ MUL}} \quad \frac{\langle \sigma', \text{bar} \rangle \downarrow \langle \sigma', 7 \rangle \text{ VAR}}{\langle \sigma', \text{foo \times bar} \rangle \downarrow \langle \sigma', 21 \rangle \text{ ASSGN}}$$

A closer look to this structure reveals the relation between small step and large-step evaluation: a depth-first traversal of the large-step proof tree yields the sequence of one-step transitions in small-step evaluation.
2 Equivalence of semantics

A natural question to ask is whether the small-step and large-step semantics are equivalent. The next theorem answers this question affirmatively.

**Theorem** (Equivalence of semantics). For all expressions \( e \), stores \( \sigma \) and \( \sigma' \), and integers \( n \) we have:

\[
\langle \sigma, e \rangle \Downarrow \langle \sigma', n \rangle \text{ if and only if } \langle \sigma, e \rangle \rightarrow^* \langle \sigma', n \rangle
\]

To streamline the proof, we will work with the following definition of the multi-step relation:

\[
\begin{align*}
\langle \sigma, e \rangle \rightarrow^* \langle \sigma, e \rangle & \quad \text{REFL} \\
\langle \sigma, e \rangle \rightarrow \langle \sigma', e' \rangle & \rightarrow^* \langle \sigma'', e'' \rangle & \quad \text{TRANS}
\end{align*}
\]

**Proof sketch.** We show each direction separately.

\( \implies \): We want to prove that the following property \( P \) holds for all expressions \( e \in \text{Exp} \):

\[
P(e) \equiv \forall \sigma, \sigma' \in \text{Store}. \forall n \in \text{Int}. \langle \sigma, e \rangle \Downarrow \langle \sigma', n \rangle \implies \langle \sigma, e \rangle \rightarrow^* \langle \sigma', n \rangle
\]

We proceed by structural induction on \( e \). We have to consider each of the possible axioms and inference rules for constructing an expression.

**Case** \( e = x \): Assume that \( \langle \sigma, x \rangle \Downarrow \langle \sigma', n \rangle \). That is, there is some derivation in the large-step operational semantics whose conclusion is \( \langle \sigma, x \rangle \Downarrow \langle \sigma, n \rangle \). There is only one rule whose conclusion matches the configuration \( \langle \sigma, x \rangle \): the large-step rule VAR. Thus, we have \( n = \sigma(x) \) and \( \sigma' = \sigma \). By the small-step rule VAR, we also have \( \langle \sigma, x \rangle \rightarrow \langle \sigma, n \rangle \). By the REF and TRANS rules, we conclude that \( \langle \sigma, x \rangle \rightarrow^* \langle \sigma, n \rangle \), which finishes the case.

**Case** \( e = n \): Assume that \( \langle \sigma, n \rangle \Downarrow \langle \sigma', n' \rangle \). There is only one rule whose conclusion matches \( \langle \sigma, n \rangle \): the large-step rule INT. Thus, we have \( n' = n \) and \( \sigma' = \sigma \) and so \( \langle \sigma, n \rangle \rightarrow^* \langle \sigma, n \rangle \) by the REF rule.

**Case** \( e = e_1 + e_2 \): This is an inductive case. We want to prove that if \( P(e_1) \) and \( P(e_2) \) hold, then \( P(e) \) also holds. Let’s write out \( P(e_1) \), \( P(e_2) \), and \( P(e) \) explicitly.

\[
\begin{align*}
P(e_1) &= \forall n, \sigma, \sigma'. \langle \sigma, e_1 \rangle \Downarrow \langle \sigma', n \rangle \implies \langle \sigma, e_1 \rangle \rightarrow^* \langle \sigma', n \rangle \\
P(e_2) &= \forall n, \sigma, \sigma'. \langle \sigma, e_2 \rangle \Downarrow \langle \sigma', n \rangle \implies \langle \sigma, e_2 \rangle \rightarrow^* \langle \sigma', n \rangle \\
P(e) &= \forall n, \sigma, \sigma'. \langle \sigma, e_1 + e_2 \rangle \Downarrow \langle \sigma', n \rangle \implies \langle \sigma, e_1 + e_2 \rangle \rightarrow^* \langle \sigma', n \rangle
\end{align*}
\]

Assume that \( P(e_1) \) and \( P(e_2) \) hold. Also assume that there exist \( \sigma, \sigma' \) and \( n \) such that \( \langle \sigma, e_1 + e_2 \rangle \Downarrow \langle \sigma', n \rangle \). We need to show that \( \langle \sigma, e_1 + e_2 \rangle \rightarrow^* \langle \sigma', n \rangle \).

We assumed that \( \langle \sigma, e_1 + e_2 \rangle \Downarrow \langle \sigma', n \rangle \). This means that there is some derivation whose conclusion is \( \langle \sigma, e_1 + e_2 \rangle \Downarrow \langle \sigma', n \rangle \). By inspection, we see that only one rule has a conclusion of this form: the ADD rule. Thus, the last rule used in the derivation was ADD and it must be the case that \( \langle \sigma, e_1 \rangle \Downarrow \langle \sigma'', n_1 \rangle \) and \( \langle \sigma'', e_2 \rangle \Downarrow \langle \sigma', n_2 \rangle \) hold for some \( n_1 \) and \( n_2 \) with \( n = n_1 + n_2 \).
By the induction hypothesis $P(e_1)$, as $\langle \sigma, e_1 \rangle \downarrow \langle \sigma'', n_1 \rangle$, we must have $\langle \sigma, e_1 \rangle \rightarrow^* \langle \sigma'', n_1 \rangle$. Likewise, by induction hypothesis $P(e_2)$, we have $\langle \sigma'', e_2 \rangle \rightarrow^* \langle \sigma', n_2 \rangle$. By Lemma 1 below, we have,

$$\langle \sigma, e_1 + e_2 \rangle \rightarrow^* \langle \sigma'', n_1 + e_2 \rangle,$$

and by another application of Lemma 1 we have:

$$\langle \sigma'', n_1 + e_2 \rangle \rightarrow^* \langle \sigma', n_1 + n_2 \rangle.$$

Then, using the small-step ADD rule and the multi-step TRANS rule, we have:

$$\frac{n = n_1 + n_2}{\langle \sigma', n_1 + n_2 \rangle \rightarrow \langle \sigma', n \rangle \quad \text{ADD}} \quad \frac{\langle \sigma', n \rangle \rightarrow^* \langle \sigma', n \rangle \quad \text{REFL}}{\langle \sigma', n_1 + n_2 \rangle \rightarrow^* \langle \sigma', n \rangle \quad \text{TRANS}}$$

Finally, by two applications of Lemma 2, we obtain $\langle \sigma, e_1 + e_2 \rangle \rightarrow^* \langle \sigma', n \rangle$, which finishes the case.

Case $e = e_1 * e_2$. Similar to case for $e_1 + e_2$ above.

Case $e = x := e_1; e_2$. Omitted. Try it as an exercise.

$\iff$: We proceed by induction on the derivation of $\langle \sigma, e \rangle \rightarrow^* \langle \sigma', n \rangle$ with a case analysis on the last rule used.

Case REFL: Then $e = n$ and $\sigma' = \sigma$. We immediately have $\langle \sigma, n \rangle \downarrow \langle \sigma, n \rangle$ by the large-step rule INT.

Case TRANS: Then $\langle \sigma, e \rangle \rightarrow \langle \sigma'', e'' \rangle$ and $\langle \sigma'', e'' \rangle \rightarrow^* \langle \sigma', n \rangle$. In this case, the induction hypothesis gives $\langle \sigma'', e'' \rangle \downarrow \langle \sigma', n \rangle$. The result follows from Lemma 3 below.

\[\square\]

Lemma 1. If $\langle \sigma, e \rangle \rightarrow^* \langle \sigma', n \rangle$, then the following hold:

- $\langle \sigma, e + e_2 \rangle \rightarrow^* \langle \sigma', n + e_2 \rangle$
- $\langle \sigma, e * e_2 \rangle \rightarrow^* \langle \sigma', n * e_2 \rangle$
- $\langle \sigma, n_1 + e \rangle \rightarrow^* \langle \sigma', n_1 + n \rangle$
- $\langle \sigma, n_1 * e \rangle \rightarrow^* \langle \sigma', n_1 * n \rangle$
- $\langle \sigma, x := e ; e_2 \rangle \rightarrow^* \langle \sigma', x := n ; e_2 \rangle$

Proof. Omitted; try it as an exercise.

\[\square\]

Lemma 2. If $\langle \sigma, e \rangle \rightarrow^* \langle \sigma', e' \rangle$ and $\langle \sigma', e' \rangle \rightarrow^* \langle \sigma'', e'' \rangle$, then $\langle \sigma, e \rangle \rightarrow^* \langle \sigma'', e'' \rangle$.

Proof. Omitted; try it as an exercise.

\[\square\]

Lemma 3. If $\langle \sigma, e \rangle \rightarrow \langle \sigma'', e'' \rangle$ and $\langle \sigma'', e'' \rangle \downarrow \langle \sigma', n \rangle$, then $\langle \sigma, e \rangle \downarrow \langle \sigma', n \rangle$.

Proof. Omitted; try it as an exercise.

\[\square\]