Elements of Programming Languages
Lecture Notes: $\mathsf{L}_{\mathsf{Let}}$

1 Abstract Syntax

$$
\begin{array}{rcll}
\mathit{Expr} \ni e & ::= & n \in \mathbb{N} \mid e_1 + e_2 \mid e_1 \times e_2 & \mathsf{L}_{\mathsf{Arith}} \\
& \mid & b \in \mathbb{B} \mid e_1 == e_2 \mid \texttt{if } e \texttt{ then } e_1 \texttt{ else } e_2 & \mathsf{L}_{\mathsf{If}} \\
& \mid & x \mid \texttt{let } x = e_1 \texttt{ in } e_2 & \mathsf{L}_{\mathsf{Let}} \\
\\
\mathit{Type} \ni \tau & ::= & \texttt{int} & \mathsf{L}_{\mathsf{Arith}} \\
& \mid & \texttt{bool} & \mathsf{L}_{\mathsf{If}} \\
\\
\Gamma & ::= & x_1 : \tau_1, \ldots, x_n : \tau_n & \\
\\
\mathit{Value} \ni v & ::= & n \in \mathbb{N} & \mathsf{L}_{\mathsf{Arith}} \\
& \mid & b \in \mathbb{B} & \mathsf{L}_{\mathsf{If}}
\end{array}
$$

1.1 Free variables

In the following, $\oplus$ stands for any binary operator.

$$
\begin{array}{rcl}
FV(n) & = & \emptyset \\
FV(e_1 \oplus e_2) & = & FV(e_1) \cup FV(e_2) \\
FV(b) & = & \emptyset \\
FV(\texttt{if } e \texttt{ then } e_1 \texttt{ else } e_2) & = & FV(e) \cup FV(e_1) \cup FV(e_2) \\
FV(x) & = & \{x\} \\
FV(\texttt{let } x = e_1 \texttt{ in } e_2) & = & FV(e_1) \cup (FV(e_2) - \{x\})
\end{array}
$$

1.2 Substitution

$$
\begin{array}{rcl}
n[e/x] & = & n \\
(e_1 \oplus e_2)[e/x] & = & e_1[e/x] \oplus e_2[e/x] \\
b[e/x] & = & b \\
(\texttt{if } e_0 \texttt{ then } e_1 \texttt{ else } e_2)[e/x] & = & \texttt{if } (e_0[e/x]) \texttt{ then } (e_1[e/x]) \texttt{ else } (e_2[e/x]) \\
x[e/x] & = & e \\
y[e/x] & = & y \quad (x \neq y) \\
(\texttt{let } y = e_1 \texttt{ in } e_2)[e/x] & = & \texttt{let } y = e_1[e/x] \texttt{ in } e_2[e/x] \\
& & (\text{where } y \# e)
\end{array}
$$

2 Evaluation

$\boxed{e \Downarrow v}$ for $\mathsf{L}_{\mathsf{Arith}}$

$$\frac{}{v \Downarrow v} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{e_1 + e_2 \Downarrow v_1 +_{\mathbb{N}} v_2} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{e_1 \times e_2 \Downarrow v_1 \times_{\mathbb{N}} v_2}$$

$\boxed{e \Downarrow v}$ for L_{If}

$$\frac{e_1 \Downarrow v \quad e_2 \Downarrow v}{e_1 == e_2 \Downarrow \texttt{true}} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2 \quad v_1 \neq v_2}{e_1 == e_2 \Downarrow \texttt{false}}$$

$$\frac{e \Downarrow \texttt{true} \quad e_1 \Downarrow v_1}{\texttt{if}\ e\ \texttt{then}\ e_1\ \texttt{else}\ e_2 \Downarrow v_1} \qquad \frac{e \Downarrow \texttt{false} \quad e_2 \Downarrow v_2}{\texttt{if}\ e\ \texttt{then}\ e_1\ \texttt{else}\ e_2 \Downarrow v_2}$$

$\boxed{e \Downarrow v}$ for $\mathsf{L}_{\mathsf{Let}}$

$$\frac{e_1 \Downarrow v_1 \quad e_2[v_1/x] \Downarrow v_2}{\texttt{let}\ x = e_1\ \texttt{in}\ e_2 \Downarrow v_2}$$

3 Types

$\boxed{\Gamma \vdash e : \tau}$ for $\mathsf{L}_{\mathsf{Arith}}$

$$\frac{}{\Gamma \vdash n : \texttt{int}} \qquad \frac{\Gamma \vdash e_1 : \texttt{int} \quad \Gamma \vdash e_2 : \texttt{int}}{\Gamma \vdash e_1 + e_2 : \texttt{int}} \qquad \frac{\Gamma \vdash e_1 : \texttt{int} \quad \Gamma \vdash e_2 : \texttt{int}}{\Gamma \vdash e_1 \times e_2 : \texttt{int}}$$

$\boxed{\Gamma \vdash e : \tau}$ for L_{If}

$$\frac{}{\Gamma \vdash b : \texttt{bool}} \qquad \frac{\Gamma \vdash e_1 : \tau \quad \Gamma \vdash e_2 : \tau}{\Gamma \vdash e_1 == e_2 : \texttt{bool}} \qquad \frac{\Gamma \vdash e : \texttt{bool} \quad \Gamma \vdash e_1 : \tau \quad \Gamma \vdash e_2 : \tau}{\Gamma \vdash \texttt{if}\ e\ \texttt{then}\ e_1\ \texttt{else}\ e_2 : \tau}$$

$\boxed{\Gamma \vdash e : \tau}$ for $\mathsf{L}_{\mathsf{Let}}$

$$\frac{\Gamma(x) = \tau}{\Gamma \vdash x : \tau} \qquad \frac{\Gamma \vdash e_1 : \tau_1 \quad \Gamma, x : \tau_1 \vdash e_2 : \tau_2}{\Gamma \vdash \texttt{let}\ x = e_1\ \texttt{in}\ e_2 : \tau_2}$$