1 Extended Grammar

Here are some additional terms not defined in the core grammar.

$$
\begin{array}{rl}
e ::= & f = \lambda x : \tau . e \\
\mid & f x \\
\mid & \texttt{val}\ x : \tau = e \\
\mid & \texttt{let}\ x = e\ \texttt{in}\ e
\end{array}
$$

2 Transformation Rules

In this section we'll show that the extended grammar can be encoded in the core grammar. To be a faithful embedding we need to show that the transformation rules preserve static and dynamic semantics. We say $e_1 \simeq e_2$ if and only if the following two holds:

- $e_1 \longrightarrow_* e_1' \mid \varepsilon' \iff e_2 \longrightarrow_* e_2' \mid \varepsilon'$
- $\forall \Gamma \mid (\Gamma \vdash e_1 : \tau\ \texttt{with}\ \varepsilon \iff \Gamma \vdash e_2 : \tau\ \texttt{with}\ \varepsilon)$

$\boxed{e_1 \simeq e_2}$

$$\frac{\Gamma \vdash e_1 : \tau_1 \quad \Gamma \vdash e_2 : \tau_2}{\texttt{let}\ y = e_1\ \texttt{in}\ e_2 \simeq (\texttt{new}\ x \Rightarrow \texttt{def}\ m(y : \tau_1) : \tau_2 = [e_1/y]e_2).m(e_2)}\ (\simeq\text{-}\textsc{Let})$$

$$\frac{\Gamma \vdash e : \tau'}{f = \lambda x : \tau . e \simeq f = \texttt{new}\ x \Rightarrow\ \texttt{def}\ m(x : \tau) : \tau' = e}\ (\simeq\text{-}\textsc{Def}\lambda)$$

$$\frac{}{f y \simeq e[y/x]}\ (\simeq\text{-}\textsc{Apply}\lambda)$$