# Day 4.

## 1. Introducing Names

Let's talk about names.

$$\mathcal{X} \ni x \qquad \mathbb{Z} \ni z \qquad \mathbb{B} \ni b \qquad \mathbb{Z} \cup \mathbb{B} \ni v$$

$$\begin{aligned}\mathcal{E} \ni e ::=& z \mid e_1 + e_2 \mid e_1 \times e_2 \\ &\mid b \mid \textsf{if } e_1 \textsf{ then } e_2 \textsf{ else } e_3 \\ &\mid x \mid \texttt{let } x = e_1 \texttt{ in } e_2\end{aligned}$$

As before, we want:

- An evaluation relation
- An approximation of the evaluation relation that guarantees safety.

What are the problems?

- $\dfrac{}{x \Downarrow ??}$
- $\dfrac{t_1 \Downarrow v_1 \quad t_2 \Downarrow v_2}{\texttt{let } x = t_1 \texttt{ in } t_2 \Downarrow v_2}$ ... but where did $v_1$ go?

## 2. Substitution

First approach: *substitute* values into terms.

We define the substitution of a expression $e$ for a variable $x$ in a term $e'$ (notation $e'[e/x]$) as follows:

$$\begin{aligned}
y[e/x] &= \begin{cases} e & \text{if } x = y \\ y & \text{otherwise} \end{cases} \\
z[e/x] &= z \\
(e_1 \odot e_2)[e/x] &= e_1[e/x] \odot e_2[e/x] && \odot \in \{+, \times\} \\
(\textsf{if } e_1 \textsf{ then } e_2 \textsf{ else } e_3)[e/x] &= \textsf{if } e_1[e/x] \textsf{ then } e_2[e/x] \textsf{ else } e_3[e/x] \\
(\texttt{let } y = e_1 \texttt{ in } e_2)[e/x] &= \begin{cases} \texttt{let } y = e_1[e/x] \texttt{ in } t_2 & \text{if } x = y \\ \texttt{let } y = e_1[e/x] \texttt{ in } e_2 ev/x] & \text{otherwise} \end{cases}
\end{aligned}$$

Relevant points:

- Shadowing of variables in `let`. (Intuition: bound names don't matter. Will pay off momentarily.)

4.

Now, we are equipped to give our first meaning of variables and `let`:

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

- Substitution is a *meta-theoretic* notion: we don't have separate evaluation rules for $x[4/x]$ and 4, we treat those as the same term.
- Relying on the inclusion of values in terms $\mathcal{V} \subseteq \mathcal{E}$. Could introduce explicit notation for this, but not even I am that pedantic.

No rule for variables:

$$\frac{\dfrac{}{4 \Downarrow 4} \quad \dfrac{\dfrac{}{4 \Downarrow 4} \quad \dfrac{}{4 \Downarrow 4}}{4 \div 4 \Downarrow 1}}{\texttt{let}\ x = 4\ \texttt{in}\ x \div x \Downarrow 1}$$

So variables are always stuck terms: no derivation for $\texttt{let}\ x = 5\ \texttt{in}\ y \Downarrow z$ for any $z$:

$$\frac{\dfrac{}{5 \Downarrow 5} \quad \overset{\lightning}{y \Downarrow z}}{\texttt{let}\ x = 5\ \texttt{in}\ y \Downarrow z}$$

## 3. $\alpha$-Equivalence

Intuition: changing the names of local variables doesn't matter. Now, we're in a position to capture this idea formally.

We define *$\alpha$-equivalence*—i.e., equivalence up to renaming of variables—by:

$$\frac{}{x \equiv_\alpha x} \qquad \frac{}{z \equiv_\alpha z} \qquad \frac{e_1 \equiv_\alpha e_1' \quad e_2 \equiv_\alpha e_2'}{e_1 \odot e_2 \equiv_\alpha e_1' \odot e_2'}\ (\odot \in \{+, \times\})$$

$$\frac{e_1 \equiv_\alpha e_1' \quad e_2[z/x] \equiv_\alpha e_2'[z/y]}{\texttt{let}\ x = e_1\ \texttt{in}\ e_2 \equiv_\alpha \texttt{let}\ y = e_1'\ \texttt{in}\ e_2'}\ (z \notin v(e_1) \cup v(e_2))$$

where the variables of an expression $v(e)$ are those variables used in a term:

$$v(x) = \{x\} \qquad v(t_1 \odot t_2) = v(t_1) \cup v(t_2), \quad \odot \in \{+, \times\}$$
$$v(z) = \emptyset \qquad v(\texttt{let}\ x = e_1\ \texttt{in}\ e_2) = v(e_1) \cup \{x\} \cup v(e_2)$$
$$v(\textsf{if}\ e_1\ \textsf{then}\ e_2\ \textsf{else}\ e_3) = v(e_1) \cup v(e_2) \cup v(e_3)$$

Why do we need a new (also called "fresh") variable in the `let` case? Mostly to avoid the possibility that $x$ is already used in $e_2'$.

Now we can make formal our intuition about $\alpha$-equivalence:

**Theorem.** *If $t \equiv_\alpha t'$ and $t \Downarrow v$ then $t' \Downarrow v$.*

*Proof.* By structural induction on the derivation of $t \equiv_\alpha t'$:

- Case $\dfrac{}{x \equiv_\alpha x}$ : the second hypothesis ($x \Downarrow v$) is impossible.

- Case $\dfrac{}{z \equiv_\alpha z}$: by definition of $\Downarrow$.
- Case $\dfrac{e_1 \equiv_\alpha e_1' \quad e_2 \equiv_\alpha e_2'}{e_1 \odot e_1' \equiv_\alpha e_2 \odot e_2'}$: If $e \Downarrow v$, then we have that $e_1 \Downarrow v_1$, $e_2 \Downarrow v_2$, and (abusing notation slightly) $v = v_1 \odot v_2$. Now, by the induction hypothesis, $e_1' \Downarrow v_1$, $e_2' \Downarrow v_2$, and finally by the definition of $\Downarrow$ we have $e' \Downarrow v$.
- The case for conditionals follows the same reasoning.
- Case $\dfrac{e_1 \equiv_\alpha e_1' \quad [z/x]e_2 \equiv_\alpha [z/y]e_2'}{\texttt{let}\ x = e_1\ \texttt{in}\ e_2 \equiv_\alpha \texttt{let}\ y = e_1'\ \texttt{in}\ e_2'}$: By the induction hypothesis applied to the first subderivation we have $e_1 \Downarrow v_1$, $e_1' \Downarrow v_1$. Similarly, by the IH applied to the second subderivation, we have $e_2[z/x][v_1/z] \Downarrow v_2$ and $e_2'[z/y][v_1/z] \Downarrow v_2$. But the latter two expressions are equivalent (because $z$ is fresh) to $e_2[v_1/x]$ and $e_2'[v_1/y]$, so we have that the original terms evaluate to $v_2$ as well. □