# Example Preservation (Subject Reduction) Proof

for a Simply-Typed Lambda Calculus

CS 6520, Spring 2000

## 1 Language

The language:

$$
\begin{array}{lcl}
t & = & x \mid \texttt{true} \mid \texttt{false} \mid \lambda x{:}T.t \mid t\ t \mid \texttt{if}\ t\ \texttt{then}\ t\ \texttt{else}\ t \\
T & = & \texttt{Bool} \mid T \to T \\
x & = & \text{a variable}
\end{array}
$$

Reductions:

$$(\lambda x{:}T.t)\ v \longrightarrow \{x \mapsto v\}t$$

$$\frac{t_1 \longrightarrow t_1'}{t_1\ t_2 \longrightarrow t_1'\ t_2} \qquad \frac{t_2 \longrightarrow t_2'}{v\ t_2 \longrightarrow v\ t_2'}$$

$$\texttt{if true then}\ t_2\ \texttt{else}\ t_3 \longrightarrow t_2$$
$$\texttt{if false then}\ t_2\ \texttt{else}\ t_3 \longrightarrow t_3$$

$$\frac{t_1 \longrightarrow t_1'}{\texttt{if}\ t_1\ \texttt{then}\ t_2\ \texttt{else}\ t_3 \longrightarrow \texttt{if}\ t_1'\ \texttt{then}\ t_2\ \texttt{else}\ t_3}$$

Type rules:

$$\Gamma \vdash \texttt{true} : \texttt{Bool} \qquad \Gamma \vdash \texttt{false} : \texttt{Bool} \qquad \Gamma \vdash x : T \ \ \text{if}\ \Gamma(x) = T$$

$$\frac{\Gamma, x{:}T \vdash t : T'}{\Gamma \vdash \lambda x{:}T.t : T \to T'}$$

$$\frac{\Gamma \vdash t_1 : T' \to T \quad \Gamma \vdash t_2 : T'}{\Gamma \vdash t_1\ t_2 : T}$$

$$\frac{\Gamma \vdash t_1 : \texttt{Bool} \quad \Gamma \vdash t_2 : T \quad \Gamma \vdash t_3 : T}{\Gamma \vdash \texttt{if}\ t_1\ \texttt{then}\ t_2\ \texttt{else}\ t_3 : T}$$

## 2 Presevation Using Mathematical Induction

Define the following size metric for expressions:

$$
\begin{array}{lcl}
|x| & = & 1 \\
|\texttt{true}| & = & 1 \\
|\texttt{false}| & = & 1 \\
|\lambda x{:}T.t| & = & 1 + |t| \\
|t_1\ t_2| & = & |t_1| + |t_2| \\
|\texttt{if}\ t_1\ \texttt{then}\ t_2\ \texttt{else}\ t_3| & = & |t_1| + |t_2| + |t_3|
\end{array}
$$

**Lemma 1 [Multi-Step Preservation]:** If $\vdash t : T$ and $t \longrightarrow^* t'$, then $\vdash t' : T$.
**Proof:** By induction on the number of $\longrightarrow$ steps showing $t \longrightarrow^* t'$.

- **Case 0 steps**
  $t' = t$, so $\vdash t' : T$.

- **Case $n$ steps (for $n > 0$)**
  $t \longrightarrow^* t'$ is $t \longrightarrow t_1 \longrightarrow \ldots t_{n-1} \longrightarrow t'$. By induction, $\vdash t_{n-1} : T$. Then, by Lemma 2 [Single-Step Preservation] and $t_{n-1} \longrightarrow t'$, $\vdash t' : T$.

$\boxed{\text{QED}}$

**Lemma 2 [Single-Step Preservation]:** If $\vdash t : T$ and $t \longrightarrow t'$, then $\vdash t' : T$.
**Proof:** By induction on $|t|$.

- **Case $|t| = 1$**
  In the three possible cases, $x$, `true`, and `false`, no reduction is possible, so the lemma holds.

- **Case $|t| = n$ (for $n > 1$)**
  We have six possible cases for the reduction $t \longrightarrow t'$.

  - **Case $(\lambda x{:}T'.t_b)\ v \longrightarrow \{x \mapsto v\}t_b$**
    Since $\vdash (\lambda x{:}T'.t_b)\ v : T$ and only one type rule applies to $t$, then it must be the case that $\vdash (\lambda x{:}T'.t_b) : T' \to T$ and $\vdash v : T'$. Furthermore, $\vdash (\lambda x{:}T'.t_b) : T' \to T$ requires that $x{:}T \vdash t_b : T$. By applying the Lemma 3 [Substitution] on $\vdash v : T'$ and $x{:}T \vdash t_b : T$, we have that $\vdash \{x \mapsto v\}t_b : T$, which is the same as $\vdash t' : T$.
  - **Case $t_1\ t_2 \longrightarrow t_1'\ t_2$**
    This reduction requires $t_1 \longrightarrow t_1'$. Also $\vdash t_1\ t_2 : T$ requires $\vdash t_1 : T' \to T$ and $\vdash t_2 : T'$ for some $T'$. Since $|t_1|$ is smaller than $|t|$, then by induction $\vdash t_1' : T' \to T$. With $\vdash t_1' : T' \to T$, $\vdash t_2 : T'$, and the type rule for application, we have $\vdash t_1'\ t_2 : T$, which is the same as $\vdash t' : T$.
  - **Case $v\ t_2 \longrightarrow v\ t_2'$**
    Analogous to the previous case.
  - **Case `if true then` $t_2$ `else` $t_3 \longrightarrow t_2$**
    Since $\vdash$ `if true then` $t_2$ `else` $t_3 : T$, then $\vdash t_2 : T$, which is the same as $\vdash t' : T$.
  - **Case `if false then` $t_2$ `else` $t_3 \longrightarrow t_3$**
    Analogous to the previous case.
  - **Case `if` $t_1$ `then` $t_2$ `else` $t_3 \longrightarrow$ `if` $t_1'$ `then` $t_2$ `else` $t_3$**
    Analogous to the $t_1\ t_2 \longrightarrow t_1'\ t_2$ case.

$\boxed{\text{QED}}$

**Lemma 3 [Substitution]:** If $\Gamma, x{:}T \vdash t : T'$ and $\vdash v : T$, then $\Gamma \vdash \{x \mapsto v\}t : T'$.
**Proof:** By induction on $|t|$.

- **Case $|t| = 1$**
  We have three possible cases for $t$:

  - **Case $x'$**
    If $x' = x$, then $\{x \mapsto v\}t$ is $v$ and $\vdash v : T$. If $x' \neq x$, then $\{x \mapsto v\}t$ is $x'$, which is $t$ by assumption, and $\vdash t : T$. In either case, $\Gamma \vdash \{x \mapsto v\}t : T'$.
  - **Case `true`**
    $\{x \mapsto v\}$`true` is still `true`, so $\Gamma \vdash \{x \mapsto v\}t : T'$.
  - **Case `false`**
    Analogous to the previous case.

- **Case $|t| = n$ (for $n > 1$)**
  We have three possible cases for $t$.

  - **Case $\lambda x_a{:}T_a.t_b$**
    If $x_a = x$, then $\{x \mapsto v\}t$ is still $t$ (by non-capturing substitution). Otherwise, $\{x \mapsto v\}t$ is $\lambda x_a{:}T_a.\{x \mapsto v\}t_b$. Since $\Gamma, x{:}T \vdash \lambda x_a{:}T_a.t_b : T'$, then $T'$ must be $T_a \to T_b$ for some $T_b$ where $\Gamma, x{:}T, x_a{:}T_a \vdash t_b : T_b$. Since $|t_b| < |t|$, we can conclude by induction that $\Gamma, x_a{:}T_a \vdash \{x \mapsto v\}t_b : T_b$. The type rule for functions then gives $\Gamma \vdash \lambda x_a{:}T_a.\{x \mapsto v\}t_b : T_a \to T_b$, which is the same as $\Gamma \vdash \{x \mapsto v\}t : T'$.
  - **Case $t_1\ t_2$**
    Since $\Gamma, x{:}T \vdash t_1\ t_2 : T'$, then $\Gamma, x{:}T \vdash t_1 : T'' \to T'$ and $\Gamma, x{:}T \vdash t_2 : T''$ for some $T''$. By induction, $\Gamma \vdash \{x \mapsto v\}t_1 : T'' \to T'$ and $\Gamma \vdash \{x \mapsto v\}t_2 : T''$. The type rule for application then gives $\Gamma \vdash \{x \mapsto v\}t_1\ \{x \mapsto v\}t_2 : T'$, which is the same as $\Gamma \vdash \{x \mapsto v\}(t_1\ t_2) : T'$, which is the same as $\Gamma \vdash \{x \mapsto v\}t : T'$.
  - **Case `if` $t_1$ `then` $t_2$ `else` $t_3$**
    Analogous to the previous case.

$\boxed{\text{QED}}$

## 3 Presevation Using Structural Induction

Since our language is defined a context-free grammar, we can use structural induction for proofs instead of mathematical induction. The shape of the proof does not change, but we avoid defining a size metric.

Instead, the proof of Lemmas 2 and 3 is "by structural indcution". Within the proof, the inductive hypothesis can be applied (with simply "by induction") to any claim about a sub-expression. A sketch for the revised proof of Lemma 2 follows.

**Lemma 2 [Single-Step Preservation]:** If $\vdash t : T$ and $t \longrightarrow t'$, then $\vdash t' : T$.
**Proof, Revised:** By structural induction on $t$.

- **Base case:**
  In the three possible cases, ... *same as before.*

- **Inductive case:**
  We have six possible cases for the reduction $t \longrightarrow t'$.

  - **Case $(\lambda x{:}T'.t_b)\ v \longrightarrow \{x \mapsto v\}t_b$**
    *Same as before.*
  - **Case $t_1\ t_2 \longrightarrow t_1'\ t_2$**
    This reduction requires $t_1 \longrightarrow t_1'$. Also $\vdash t_1\ t_2 : T$ requires $\vdash t_1 : T' \to T$ and $\vdash t_2 : T'$ for some $T'$. By induction, $\vdash t_1' : T' \to T$. With $\vdash t_1' : T' \to T$, $\vdash t_2 : T'$, and the type rule for application, we have $\vdash t_1'\ t_2 : T$, which is the same as $\vdash t' : T$.
  - **Case $v\ t_2 \longrightarrow v\ t_2'$**

    *The rest is the same as before...*

$\boxed{\text{QED}}$