CS 578 Programming Language Semantics – Spring 2022

Mystery Problem

Recall from the midterm the simply typed $\lambda$-calculus extended with Booleans and Pairs, which will be denoted $\lambda_{\rightarrow,B,\times}$. This problem concerns the big-step evaluation rules for $\lambda_{\rightarrow,B,\times}$. Consider the following theorem and accompanying “proof.”

**Theorem.** If $\vdash t : T$, then there exists a $v$ such that $t \downarrow v$ and $\vdash v : T$.

**Proof.** By induction on the typing derivation $\vdash t : T$. We proceed by case analysis on the final rule in the derivation.

- Case T-VAR: This case is impossible, since the typing environment is empty.
- Case T-ABS: In this case $t$ has the form $\lambda x : T_1 . t_1$, so it is already a value, and the conclusion is immediate.
- Case T-APP: $t = t_1 t_2$
  \[ \begin{array}{l}
  \vdash t_1 : T_1 \rightarrow T_2 \\
  \vdash t_2 : T_1 \\
  T = T_2 
  \end{array} \]
  By induction on the derivation for $t_1$, there exists a value $v_1$ such that $t_1 \downarrow v_1$ and $\vdash v_1 : T_1 \rightarrow T_2$. By the Canonical Forms Lemma, $v$ can be written as $\lambda x : T_1 . t_{11}$. Hence we must have $x : T_1 \vdash t_{11} : T_2$. By induction on the derivation for $t_2$, there exists a $v_2$ such that $t_2 \downarrow v_2$ and $\vdash v_2 : T_1$. By the Substitution Lemma, $\vdash [x \mapsto v_2] t_{11} : T_2$. Finally, by induction on this derivation, there exists a $v$ such that $[x \mapsto v_2] t_{11} \downarrow v$ and $\vdash v : T_2$. Combining our results and applying B-APP, we get $t_1 t_2 \downarrow v$ and $\vdash v : T$, as required.
- Case T-TRUE: Similar to T-ABS.
- Case T-FALSE: Similar to T-ABS.
- Case T-IF: $t = \text{if } t_1 \text{ then } t_2 \text{ else } t_3$
  \[ \begin{array}{l}
  \vdash t_1 : \text{Bool} \\
  \vdash t_2 : T \\
  \vdash t_3 : T 
  \end{array} \]
  By induction on the derivation for $t_1$, there exists a value $v_1$ such that $t_1 \downarrow v_1$ and $\vdash v_1 : \text{Bool}$. By the Canonical Forms Lemma, $v_1$ must be either true or false. Suppose $v_1 = \text{true}$ (the other case is similar). By induction on the derivation for $t_2$, there exists a $v_2$ such that $t_2 \downarrow v_2$ and $\vdash v_2 : T$. By B-IFTRUE, $t \downarrow v_2$, as required.
- Case T-PAIR: $t = \{t_1, t_2\}$
  \[ \begin{array}{l}
  \vdash t_1 : T_1 \\
  \vdash t_2 : T_2 \\
  T = T_1 \times T_2 
  \end{array} \]
  By induction on the derivation for $t_1$, there exists a $v_1$ such that $t_1 \downarrow v_1$ and $\vdash v_1 : T_1$. Similarly, there exists a $v_2$ such that $t_2 \downarrow v_2$ and $\vdash v_2 : T_2$. Then $v = \{v_1, v_2\}$ is a value. By B-PAIR, $t \downarrow v$ and by T-PAIR, $\vdash v : T_1 \times T_2$, as required.
• Case T-PROJ1: $t = t_0 \cdot 1$
  $\vdash t_0 : T_1 \times T_2$
  $T = T_1$

By induction on the derivation for $t_0$, there exists a $v_0$ such that $t_0 \Downarrow v_0$ and $\vdash v_0 : T_1 \times T_2$.
By the Canonical Forms Lemma, $v_0$ must be of the form $\{v_1, v_2\}$ for some values $v_1$ and $v_2$. By B-PROJ1, $t \Downarrow v_1$, and, by inversion, $\vdash v_1 : T_1$, as required.

• Case T-PROJ2: Similar to T-PROJ1.

□

This theorem is actually true, but the “proof” has a fatal flaw. Identify precisely where and how the proof goes wrong.

Rules for Simply Typed $\lambda$-calculus with Booleans and Pairs ($\lambda_{\rightarrow,\text{B},\times}$)

Typing rules

\[
\frac{x : T \in \Gamma}{\Gamma \vdash x : T} \quad \text{(T-VAR)}
\]

\[
\frac{\Gamma, x : T_1 \vdash t_2 : T_2}{\Gamma \vdash \lambda x : T_1. t_2 : T_1 \to T_2} \quad \text{(T-ABS)}
\]

\[
\frac{\Gamma \vdash t_1 : T_{11} \to T_{12} \quad \Gamma \vdash t_2 : T_{11}}{\Gamma \vdash t_1 \, t_2 : T_{12}} \quad \text{(T-APP)}
\]

\[
\Gamma \vdash \text{true} : \text{Bool} \quad \Gamma \vdash \text{false} : \text{Bool} \quad \text{(T-TRUE)} \quad \text{(T-FALSE)}
\]

\[
\frac{\Gamma \vdash t_1 : \text{Bool} \quad \Gamma \vdash t_2 : T \quad \Gamma \vdash t_3 : T}{\Gamma \vdash \text{if } t_1 \text{ then } t_2 \text{ else } t_3 : T} \quad \text{(T-IF)}
\]

\[
\frac{\Gamma \vdash t_1 : T_1 \quad \Gamma \vdash t_2 : T_2}{\Gamma \vdash \{t_1, t_2\} : T_1 \times T_2} \quad \text{(T-PAIR)}
\]

\[
\frac{\Gamma \vdash t_1 : T_{11} \times T_{12}}{\Gamma \vdash t_1.1 : T_{11}} \quad \text{(T-PROJ1)}
\]

\[
\frac{\Gamma \vdash t_1 : T_{11} \times T_{12}}{\Gamma \vdash t_1.2 : T_{12}} \quad \text{(T-PROJ2)}
\]

Big-step evaluation rules

\[
\frac{v \Downarrow v}{v \Downarrow v} \quad \text{(B-VALUE)}
\]

\[
\frac{t_1 \Downarrow (\lambda x : T_{11}. t_{12}) \quad t_2 \Downarrow v_2 \quad [x \mapsto v_2]t_{12} \Downarrow v}{t_1 \, t_2 \Downarrow v} \quad \text{(B-APP)}
\]

\[
\frac{t_1 \Downarrow \text{true} \quad t_2 \Downarrow v_2}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow v_2} \quad \text{(B-IfTRUE)}
\]

\[
\frac{t_1 \Downarrow \text{false} \quad t_3 \Downarrow v_3}{\text{if } t_1 \text{ then } t_2 \text{ else } t_3 \Downarrow v_3} \quad \text{(B-IfFALSE)}
\]

\[
\frac{t_1 \Downarrow v_1 \quad t_2 \Downarrow v_2 \quad \{t_1, t_2\} \Downarrow \{v_1, v_2\}}{t.1 \Downarrow v_1} \quad \text{(B-PROJ1)}
\]

\[
\frac{t_1 \Downarrow v_1 \quad t_2 \Downarrow v_2 \quad \{t_1, t_2\} \Downarrow \{v_1, v_2\}}{t.2 \Downarrow v_2} \quad \text{(B-PROJ2)}
\]