CS 578 Programming Language Semantics – Mid-term Exam Sample Solutions

All the questions concern the simply typed λ-calculus extended with Booleans and Pairs, which will be denoted $\lambda_{\to,\mathbb{B},\times}$.

1. [15 pts.]

Consider the term

$$\texttt{t} = (\lambda\texttt{y}:(\texttt{Bool} \to \texttt{Bool}) \times \texttt{Bool}.(\texttt{y.1})(\texttt{y.2}))\ \{\lambda\texttt{x}:\texttt{Bool.x}, \texttt{false}\}$$

When answering the following questions, you may abbreviate `Bool` by `B` and `false` by `F` to save writing time.

(a) [5 pts.] Show the sequence of one-step evaluation reductions that lead from `t` to the normal form `false`. It is *not* necessary to give the full derivation for each transition. (Hint: Four steps are needed.)

Answer:

$$\begin{aligned}
&(\lambda\texttt{y}:(\texttt{B} \to \texttt{B}) \times \texttt{B}.(\texttt{y.1})(\texttt{y.2}))\ \{\lambda\texttt{x}:\texttt{B.x},\texttt{F}\}\\
&\to (\{\lambda\texttt{x}:\texttt{B.x},\texttt{F}\}\texttt{.1})(\{\lambda\texttt{x}:\texttt{B.x},\texttt{F}\}\texttt{.2})\\
&\to (\lambda\texttt{x}:\texttt{B.x})(\{\lambda\texttt{x}:\texttt{B.x},\texttt{F}\}\texttt{.2})\\
&\to (\lambda\texttt{x}:\texttt{B.x})\ \texttt{F}\\
&\to \texttt{F}
\end{aligned}$$

Note that labeling the steps with single E-rule names is inappropriate in general, because each step represents an entire derivation built out of (potentially many) E-rules. Although specifically not required by the question, it would also be fine to write down each of these derivations in full, in which case each rule use could be labeled.

(b) [10 pts.] Draw a derivation tree using the typing rules to show that $\vdash \texttt{t} : \texttt{B}$. (Hint: Your tree should have 11 nodes.)

Answer:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\texttt{y} : (\texttt{B}\to\texttt{B})\times\texttt{B} \in \Gamma_y}{\Gamma_y \vdash \texttt{y} : (\texttt{B}\to\texttt{B})\times\texttt{B}}\ \text{T-VAR}}{\Gamma_y \vdash \texttt{y.1} : \texttt{B}\to\texttt{B}}\ \text{T-PROJ1} \qquad \dfrac{\dfrac{\texttt{y} : (\texttt{B}\to\texttt{B})\times\texttt{B} \in \Gamma_y}{\Gamma_y \vdash \texttt{y} : (\texttt{B}\to\texttt{B})\times\texttt{B}}\ \text{T-VAR}}{\Gamma_y \vdash \texttt{y.2} : \texttt{B}}\ \text{T-PROJ2}}{\Gamma_y \vdash (\texttt{y.1})(\texttt{y.2}) : \texttt{B}}\ \text{T-APP}}{\vdash (\lambda\texttt{y}:(\texttt{B}\to\texttt{B})\times\texttt{B}.(\texttt{y.1})(\texttt{y.2})) : ((\texttt{B}\to\texttt{B})\times\texttt{B})\to\texttt{B}}\ \text{T-ABS} \qquad \dfrac{\dfrac{\dfrac{\texttt{x} : \texttt{B} \in \texttt{x} : \texttt{B}}{\texttt{x} : \texttt{B} \vdash \texttt{x} : \texttt{B}}\ \text{T-VAR}}{\vdash \lambda\texttt{x}:\texttt{B.x} : \texttt{B}\to\texttt{B}}\ \text{T-ABS} \qquad \dfrac{}{\vdash \texttt{F} : \texttt{B}}\ \text{T-FALSE}}{\vdash \{\lambda\texttt{x}:\texttt{B.x},\texttt{F}\} : (\texttt{B}\to\texttt{B})\times\texttt{B}}\ \text{T-PAIR}}{\vdash (\lambda\texttt{y}:(\texttt{B}\to\texttt{B})\times\texttt{B}.(\texttt{y.1})(\texttt{y.2}))\ \{\lambda\texttt{x}:\texttt{B.x},\texttt{F}\} : \texttt{B}}\ \text{T-APP}$$

where $\Gamma_y = \texttt{y} : (\texttt{B} \to \texttt{B}) \times \texttt{B}$.

2. [20 pts.]

Consider the following meta-properties that may apply to a language.

- **Determinacy** (of one-step evaluation): If $\mathtt{t} \to \mathtt{t}'$ and $\mathtt{t} \to \mathtt{t}''$ then $\mathtt{t}' = \mathtt{t}''$.
- **Uniqueness** (of normal forms): If $\mathtt{t} \to^* \mathtt{u}$ and $\mathtt{t} \to^* \mathtt{u}'$, where $\mathtt{u}$ and $\mathtt{u}'$ are both normal forms, then $\mathtt{u} = \mathtt{u}'$.
- **Termination** (of evaluation): For every term $\mathtt{t}$ there is some normal form $\mathtt{t}'$ such that $\mathtt{t} \to^* \mathtt{t}'$.
- **Progress**: If $\vdash \mathtt{t} : \mathtt{T}$ then either $\mathtt{t}$ is a value or else $\exists \mathtt{t}'$ such that $\mathtt{t} \to \mathtt{t}'$.
- **Preservation**: If $\vdash \mathtt{t} : \mathtt{T}$ and $\mathtt{t} \to \mathtt{t}'$ then $\vdash \mathtt{t}' : \mathtt{T}$.

For each of the following languages, state which, if any, of the properties are **false**, and, for each such property, give a brief counter-example demonstrating that the property does not hold.

(a) [5 pts.] Language $\lambda_{\to,\mathbb{B},\times}$.

Answer:

- Termination. Counter-example: $(\lambda \mathtt{x}\!:\!\mathtt{Bool}.\mathtt{x}\ \mathtt{x})(\lambda \mathtt{x}\!:\!\mathtt{Bool}.\mathtt{x}\ \mathtt{x})$. Of course, this term is not well-typed, but that's irrelevant.

(b) [5 pts.] Language $\lambda_{\to,\mathbb{B},\times}$ with the addition of a small-step rule

$$\{\mathtt{v}_1, \mathtt{v}_2\}.\mathtt{1} \to \mathtt{v}_2 \qquad \text{(E-FUNNY1)}$$

Answer:

- Determinacy. Counter-example: Take $\mathtt{t} = \{\mathtt{true}, \mathtt{false}\}.\mathtt{1}$. By E-PAIRBETA1, $\mathtt{t} \to \mathtt{true}$, but by E-FUNNY1, $\mathtt{t} \to \mathtt{false}$.
- Uniqueness. Since $\mathtt{true}$ and $\mathtt{false}$ are normal forms, same counter-example works.
- Preservation. Counter-example: Take $\mathtt{t} = \{\lambda \mathtt{x}\!:\!\mathtt{Bool}.\mathtt{x}, \mathtt{true}\}.\mathtt{1}$. We have $\vdash \mathtt{t} : \mathtt{Bool} \to \mathtt{Bool}$. But by E-FUNNY1, $\mathtt{t} \to \mathtt{true}$, and $\vdash \mathtt{true} : \mathtt{Bool}$.
- Termination. As in part (a).

(c) [5 pts.] Language $\lambda_{\to,\mathbb{B},\times}$ with the addition of a small-step rule

$$\frac{\mathtt{t}_2 \to \mathtt{t}_2'}{\{\mathtt{t}_1, \mathtt{t}_2\} \to \{\mathtt{t}_1, \mathtt{t}_2'\}} \qquad \text{(E-FUNNY2)}$$

Answer:

- Determinacy. Counter-example: Take $\texttt{t} = \{(\lambda\texttt{x:Bool.x})\texttt{true},(\lambda\texttt{x:Bool.x})\texttt{true}\}$. By E-PAIR1, $\texttt{t} \to \{\texttt{true},(\lambda\texttt{x:Bool.x})\texttt{true}\}$, but by E-FUNNY2, $\texttt{t} \to \{(\lambda\texttt{x:Bool.x})\texttt{true},\texttt{true}\}$. Note, though, that Uniqueness is still true.

- Termination. As in part (a).

(d) [5 pts.] Language $\lambda_{\to,\mathbb{B},\times}$ with the addition of the typing rule

$$\frac{\Gamma \vdash \texttt{t}_1 : \texttt{Bool} \qquad \Gamma \vdash \texttt{t}_2 : \texttt{Bool}}{\Gamma \vdash \{\texttt{t}_1,\texttt{t}_2\} : \texttt{Bool}} \tag{T-FUNNY3}$$

Answer:

- Progress. Counter-example: Take $\texttt{t} = \texttt{if }\{\texttt{true},\texttt{false}\}\texttt{ then true else false}$. Then, using T-FUNNY3 we can show $\vdash \texttt{t} : \texttt{Bool}$, but $\texttt{t}$ is stuck.

- Termination. As in part (a).

Note that Preservation is *not* invalidated.

3. [25 pts.] The following Preservation theorem holds for $\lambda_{\to,\mathbb{B},\times}$:

Theorem. If $\vdash \texttt{t} : \texttt{T}$ and $\texttt{t} \to \texttt{t}'$, then $\vdash \texttt{t}' : \texttt{T}$.

An incomplete proof of this theorem is given below. Complete the proof by filling in the three missing cases (marked by a ?). You may assume the following lemmas without proof:

- **Substitution Lemma**: If $\Gamma, \texttt{x} : \texttt{S} \vdash \texttt{t} : \texttt{T}$ and $\Gamma \vdash \texttt{s} : \texttt{S}$, then $\Gamma \vdash [\texttt{x} \mapsto \texttt{s}]\texttt{t} : \texttt{T}$.

- **Inversion Lemma** of the usual form.

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

- Case T-VAR:

  Answer:

  $\texttt{t} = \texttt{z}$. Since the context is empty, the premise of this rule can never be true, so this case cannot occur. Alternatively, we can argue that no one-step rule applies, so this case cannot occur for that reason too.

- Case T-ABS: No one-step rule applies, so this case cannot occur.

- Case T-APP:

  Answer:

  $\texttt{t} = \texttt{t}_1\ \texttt{t}_2$
  $\vdash \texttt{t}_1 : \texttt{T}_{11} \to \texttt{T}_{12}$
  $\vdash \texttt{t}_2 : \texttt{T}_{11}$
  $\texttt{T} = \texttt{T}_{12}$

  There are three cases, based on the possible one-step rules.

- E-APP1: Here $t_1 \to t_1'$ and $t' = t_1'\ t_2$. By induction, $\vdash t_1' : T_{11} \to T_{12}$. By T-APP, $\vdash t' : T_{12}$.
- E-APP2: Here $t_2 \to t_2'$, and $t' = t_1 t_2'$. By induction, $\vdash t_2' : T_{11}$. By T-APP, $\vdash t' : T_{12}$.
- E-APPABS: Here $t_1 = \lambda x{:}T_{11}.t_{12}$, $t_2 = v_2$, and $t' = [x \mapsto v_2]t_{12}$. By Inversion Lemma, $x : T_{11} \vdash t_{12} : T_{12}$. By Substitution Lemma, $\vdash [x \mapsto v_2]t_{12} : T_{12}$.

• Case T-TRUE: No one-step rule applies, so this case cannot occur.

• Case T-FALSE: Similar to T-TRUE.

• Case T-IF: $t = \texttt{if}\ t_1\ \texttt{then}\ t_2\ \texttt{else}\ t_3$
$\vdash t_1 : \texttt{Bool}$
$\vdash t_2 : T$
$\vdash t_3 : T$

There are three cases, based on the possible one-step rules.

- E-IFTRUE: Here $t_1 = \texttt{true}$ and $t' = t_2$, so result is immediate.
- E-IFFALSE: Similar.
- E-IF: Here $t_1 \to t_1'$ and $t' = \texttt{if}\ t_1'\ \texttt{then}\ t_2\ \texttt{else}\ t_3$. By induction, $\vdash t_1' : \texttt{Bool}$. By T-IF, $\vdash t' : T$.

• Case T-PAIR: $t = \{t_1, t_2\}$
$\vdash t_1 : T_1$
$\vdash t_2 : T_2$
$T = T_1 \times T_2$

There are two cases, based on the possible one-step rules

- E-PAIR1: Here $t_1 \to t_1'$ and $t' = \{t_1', t_2\}$. By induction, $\vdash t_1' : T_1$. By T-PAIR, $\vdash \{t_1', t_2\} : T_1 \times T_2$.
- E-PAIR2: Similar.

• Case T-PROJ1:

Answer:

$t = t_1.1$
$\vdash t_1 : T_{11} \times T_{12}$
$T = T_{11}$

There are two cases, based on the possible one-step rules

- E-PAIRBETA1: Here $t_1 = \{v_1, v_2\}$ and $t' = v_1$. By Inversion Lemma, $\vdash v_1 : T_{11}$.
- E-PROJ1 Here $t_1 \to t_1'$ and $t' = t_1'.1$. By induction, $\vdash t_1' : T_{11} \times T_{12}$. By T-PROJ1, $\vdash t_1'.1 : T_{11}$.

• Case T-PROJ2: Similar to T-PROJ1.

4. [15 pts.]

This question asks about other semantic presentations of $\lambda_{\to,\mathbb{B},\times}$.

(a) [5 pts.] Here is a partial set of **big-step** evaluation rules for $\lambda_{\to,\mathbb{B},\times}$. Add the three missing rules.

$$\mathtt{v} \Downarrow \mathtt{v} \qquad \text{(B-VALUE)}$$

$$\frac{\mathtt{t}_1 \Downarrow (\lambda \mathtt{x}\mathtt{:}\mathtt{T}_{11}\mathtt{.}\mathtt{t}_{12}) \qquad \mathtt{t}_2 \Downarrow \mathtt{v}_2 \qquad [\mathtt{x} \mapsto \mathtt{v}_2]\mathtt{t}_{12} \Downarrow \mathtt{v}}{\mathtt{t}_1\ \mathtt{t}_2 \Downarrow \mathtt{v}} \qquad \text{(B-APP)}$$

$$\frac{\mathtt{t}_1 \Downarrow \mathtt{true} \qquad \mathtt{t}_2 \Downarrow \mathtt{v}_2}{\mathtt{if}\ \mathtt{t}_1\ \mathtt{then}\ \mathtt{t}_2\ \mathtt{else}\ \mathtt{t}_3 \Downarrow \mathtt{v}_2} \qquad \text{(B-IFTRUE)}$$

$$\frac{\mathtt{t}_1 \Downarrow \mathtt{false} \qquad \mathtt{t}_3 \Downarrow \mathtt{v}_3}{\mathtt{if}\ \mathtt{t}_1\ \mathtt{then}\ \mathtt{t}_2\ \mathtt{else}\ \mathtt{t}_3 \Downarrow \mathtt{v}_3} \qquad \text{(B-IFFALSE)}$$

Answer:

$$\frac{\mathtt{t}_1 \Downarrow \mathtt{v}_1 \qquad \mathtt{t}_2 \Downarrow \mathtt{v}_2}{\{\mathtt{t}_1,\mathtt{t}_2\} \Downarrow \{\mathtt{v}_1,\mathtt{v}_2\}} \qquad \text{(B-PAIR)}$$

$$\frac{\mathtt{t} \Downarrow \{\mathtt{v}_1,\mathtt{v}_2\}}{\mathtt{t.1} \Downarrow \mathtt{v}_1} \qquad \text{(B-PROJ1)}$$

$$\frac{\mathtt{t} \Downarrow \{\mathtt{v}_1,\mathtt{v}_2\}}{\mathtt{t.2} \Downarrow \mathtt{v}_2} \qquad \text{(B-PROJ2)}$$

(b) [5 pts.] Now consider a **contextual** semantics for $\lambda_{\to,\mathbb{B},\times}$. As usual, the top-level evaluation relation is given by a single rule

$$\frac{\mathtt{t} \to_{cmp} \mathtt{t}'}{\mathtt{C}[\mathtt{t}] \to_{ctx} \mathtt{C}[\mathtt{t}']} \qquad \text{(E-STEP)}$$

The computation rules $\to_{cmp}$ are just a subset of the small-step rules. Which ones? (Just list their names.)

Answer:

E-APPABS,E-IFTRUE,E-IFFALSE,E-PAIRBETA1,E-PAIRBETA2.

(c) [5 pts.] Still considering the **contextual** semantics for $\lambda_{\to,\mathbb{B},\times}$, complete this definition for the grammar of contexts:

Answer:

$$\mathtt{C} ::= [\ ] \mid \mathtt{C}\mathtt{t} \mid \mathtt{v}\mathtt{C} \mid \mathtt{if}\ \mathtt{C}\ \mathtt{then}\ \mathtt{t}_2\ \mathtt{else}\ \mathtt{t}_3 \mid \mathtt{C.1} \mid \mathtt{C.2} \mid \{\mathtt{C},\mathtt{t}\} \mid \{\mathtt{v},\mathtt{C}\}$$