CIS 500 Software Foundations

Homework Assignment 7

References

Due: Monday, November 8, 2004, by noon

Submission instructions: Same as last time.

1 Exercise What should be put in the `...` in the following expressions written in the language of chapter 13 (augmented with let, unit and multiplication) so that each expression evaluates to 42?

1. ```
   let r = ref 41  in
   let x = ...  in
      !r
   ```

2. ```
   let r = ref 41 in
   let x = (λr:ref Nat. (r := 41; 500)) (...)  in
      !r
   ```

3. ```
   let f = ... in
      (f ()) * (f ())
   end
   ```

2 Exercise Definition 13.5.1 describes when a store μ is well-typed with respect to a typing context Γ and a store typing Σ. What happens if we change this definition to be described by the following two rules:

$$\Gamma \mid \emptyset \vdash \emptyset$$

$$\frac{\Gamma \mid \Sigma \vdash \mu \qquad \Gamma \mid \Sigma \vdash \mathtt{v} : \mathtt{T}}{\Gamma \mid \Sigma, \mathtt{l} : \mathtt{T} \vdash \mu, \mathtt{l} = \mathtt{v}}$$

Do any of the theorems/lemmas 13.5.3-13.5.7 break with this new definition?

3 Exercise 13.5.8 in TAPL

4 Exercise Another way to formalize mutation in the simply-typed lambda calculus is with an environment-based operational semantics. Here, in function application, instead of using substitution to immediately replace the values of the bound variables, mappings for those variables are stored in a finite map called an environment. In many ways, environments are similar to heaps. However, unlike heaps, which are finite maps from store locations to values, environments are finite maps from variables to values.

An important feature of this calculus is that lambda expressions, $\lambda\mathtt{x} : \mathtt{T}.\mathtt{t}$, are not values. Instead, these expressions immediately step to closures: $\mathtt{cl}(\eta \mid \mathtt{x} : \mathtt{T}.\mathtt{t})$, values that represent not only the function but also the current environment η. Closures make sure that we always have a mapping for any of the variables in the scope of the function.
   ```

Syntax

$$
\begin{aligned}
\texttt{t} ::= \;& \texttt{x} \\
& \lambda\texttt{x}:\texttt{T.t} \\
& \texttt{cl}(\eta \mid \texttt{x}:\texttt{T.t}) \\
& \texttt{t t} \\
& \texttt{t};\texttt{t} \\
& \texttt{x} := \texttt{t} \\
& \texttt{unit}
\end{aligned}
$$

$$
\begin{aligned}
\texttt{v} ::= \;& \texttt{unit} \\
& \texttt{cl}(\eta \mid \texttt{x}:\texttt{T.t})
\end{aligned}
$$

$$
\begin{aligned}
\eta ::= \;& \emptyset \\
& \eta, \texttt{x} = \texttt{v}
\end{aligned}
$$

$$
\begin{aligned}
\texttt{T} ::= \;& \texttt{Unit} \\
& \texttt{T} \to \texttt{T}
\end{aligned}
$$

Evaluation rules

$$\texttt{x} \mid \eta \longrightarrow \eta(\texttt{x}) \mid \eta$$

$$\lambda\texttt{x}:\texttt{T.t} \mid \eta \longrightarrow \texttt{cl}(\eta \mid \texttt{x}:\texttt{T.t}) \mid \eta$$

$$\frac{\texttt{x} \notin dom(\eta')}{\texttt{cl}(\eta' \mid \texttt{x}:\texttt{T.t})\ \texttt{v} \mid \eta \longrightarrow \texttt{t} \mid \eta', \texttt{x} = \texttt{v}}$$

$$\texttt{x} := \texttt{v} \mid \eta \longrightarrow \texttt{unit} \mid [\texttt{x} \mapsto \texttt{v}]\eta$$

$$\frac{\texttt{t}_1 \mid \eta \longrightarrow \texttt{t}_1' \mid \eta'}{\texttt{t}_1\ \texttt{t}_2 \mid \eta \longrightarrow \texttt{t}_1'\ \texttt{t}_2 \mid \eta'}$$

$$\frac{\texttt{t}_2 \mid \eta \longrightarrow \texttt{t}_2' \mid \eta'}{\texttt{v}\ \texttt{t}_2 \mid \eta \longrightarrow \texttt{v}\ \texttt{t}_2 \mid \eta'}$$

$$\frac{\texttt{t}_1 \mid \eta \longrightarrow \texttt{t}_1' \mid \eta'}{\texttt{t}_1;\texttt{t}_2 \mid \eta \longrightarrow \texttt{t}_1';\texttt{t}_2 \mid \eta'}$$

$$\texttt{unit};\texttt{t}_2 \mid \eta \longrightarrow \texttt{t}_2 \mid \eta'$$

$$\frac{\texttt{t}_1 \mid \eta \longrightarrow \texttt{t}_1' \mid \eta'}{\texttt{x} := \texttt{t}_1 \mid \eta \longrightarrow \texttt{x} := \texttt{t}_1' \mid \eta'}$$

1. What is the definition of a well-typed environment $\Gamma \vdash \eta$? Note, we do not need store typings Σ because we do not have store locations. Only variables are used in environments.
2. What are the typing rules for this calculus?
3. Using your typing rules, give a typing derivation with the empty context for the term:

$$\lambda\texttt{x}:\texttt{Unit.cl}(\texttt{z} = \texttt{cl}(\emptyset|\texttt{v}:\texttt{Unit.v})|\ \texttt{w}:\texttt{Unit.w} := \texttt{zw})\ \texttt{x}$$

4. Which of the following is the correct statement of the preservation lemma for this calculus? For the incorrect statements, say which case of the proof breaks.
   (a) If $\Gamma \vdash \texttt{t}:\texttt{T}$ and $\Gamma \vdash \eta$ and $\texttt{t} \mid \eta \longrightarrow \texttt{t}' \mid \eta'$ then $\Gamma \vdash \texttt{t}':\texttt{T}$ and $\Gamma \vdash \eta'$.
   (b) If $\Gamma \vdash \texttt{t}:\texttt{T}$ and $\Gamma' \vdash \eta$ and $\texttt{t} \mid \eta \longrightarrow \texttt{t}' \mid \eta'$ then $\Gamma \vdash \texttt{t}':\texttt{T}$ and $\Gamma' \vdash \eta'$.
   (c) If $\Gamma \vdash \texttt{t}:\texttt{T}$ and $\Gamma \vdash \eta$ and $\texttt{t} \mid \eta \longrightarrow \texttt{t}' \mid \eta'$ then for some Γ', $\Gamma' \vdash \texttt{t}':\texttt{T}$ and $\Gamma' \vdash \eta'$.
   (d) If $\Gamma \vdash \texttt{t}:\texttt{T}$ and $\Gamma \vdash \eta$ and $\texttt{t} \mid \eta \longrightarrow \texttt{t}' \mid \eta'$ then for some $\Gamma' \supseteq \Gamma$, $\Gamma' \vdash \texttt{t}':\texttt{T}$ and $\Gamma' \vdash \eta'$.

5 Debriefing

1. How many hours (per person) did you spend on this assignment?
2. Would you rate it as easy, moderate, or difficult?
3. Did everyone in your study group participate?
4. How deeply do you feel you understand the material it covers (0%–100%)?

If you have any other comments, we would like to hear them; please send them to cis500@cis.upenn.edu.