C.S. 430 Assignment 6, Sample Solutions

Paul Liu

November 15, 2007

Note that these are *sample* solutions only; in many cases there were many acceptable answers.

1 Reynolds Problem 10.1

1.1 Normal-order Reductions

$$
\begin{array}{ll}
(\lambda f.\, \lambda x.\, f\,(f\,x))\,(\lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x)\,(\lambda z.\, \lambda w.\, z) \Rightarrow & \\
(\lambda x.\, (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,x))\,(\lambda z.\, \lambda w.\, z) \Rightarrow & \\
(\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,(\lambda z.\, \lambda w.\, z)) \Rightarrow & \\
\lambda x'.\, \lambda y.\, ((\lambda b.\, \lambda x''.\, \lambda y'.\, b\,y'\,x'')\,(\lambda z.\, \lambda w.\, z))\,y\,x' \Rightarrow & \text{(canonical form)} \\
\lambda x'.\, \lambda y.\, (\lambda x''.\, \lambda y'.\, (\lambda z.\, \lambda w.\, z)\,y'\,x'')\,y\,x' \Rightarrow & \\
\lambda x'.\, \lambda y.\, (\lambda y'.\, (\lambda z.\, \lambda w.\, z)\,y'\,y)\,x' \Rightarrow & \\
\lambda x'.\, \lambda y.\, (\lambda z.\, \lambda w.\, z)\,x'\,y \Rightarrow & \\
\lambda x'.\, \lambda y.\, (\lambda w.\, x')\,y \Rightarrow & \\
\lambda x'.\, \lambda y.\, x' = & \\
TRUE &
\end{array}
$$

1.2 Eager Reductions

$$
\begin{array}{ll}
(\lambda f.\, \lambda x.\, f\,(f\,x))\,(\lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x)\,(\lambda z.\, \lambda w.\, z) \Rightarrow & \\
(\lambda x.\, (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,x))\,(\lambda z.\, \lambda w.\, z) \Rightarrow & \\
(\lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x)\,((\lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x)\,(\lambda z.\, \lambda w.\, z)) \Rightarrow & \\
(\lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x)\,(\lambda x.\, \lambda y.\, (\lambda z.\, \lambda w.\, z)\,y\,x) \Rightarrow & \\
\lambda x.\, \lambda y.\, (\lambda x'.\, \lambda y'.\, (\lambda z.\, \lambda w.\, z)\,y'\,x')\,y\,x) \Rightarrow & \text{(canonical form)} \\
\lambda x.\, \lambda y.\, (\lambda y'.\, (\lambda z.\, \lambda w.\, z)\,y'\,y)\,x \Rightarrow & \\
\lambda x.\, \lambda y.\, (\lambda z.\, \lambda w.\, z)\,x\,y \Rightarrow & \\
\lambda x.\, \lambda y.\, (\lambda w.\, x)\,y \Rightarrow & \\
\lambda x.\, \lambda y.\, x = & \\
TRUE &
\end{array}
$$

1.3 Proofs of Normal-order Evaluations

We use the indented proof style.

$$
\begin{array}{l}
(\lambda f.\, \lambda x.\, f\,(f\,x))\,(\lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x)\,(\lambda z.\, \lambda w.\, z) \\
\quad (\lambda f.\, \lambda x.\, f\,(f\,x))\,(\lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x) \\
\quad\quad \lambda f.\, \lambda x.\, f\,(f\,x) \Rightarrow \lambda f.\, \lambda x.\, f\,(f\,x) \\
\quad\quad (\lambda x.\, (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,x)) \Rightarrow (\lambda x.\, (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,x)) \\
\quad \Rightarrow (\lambda x.\, (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,x)) \\
\quad (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,(\lambda z.\, \lambda w.\, z)) \\
\quad\quad \lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x' \Rightarrow \lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x' \\
\quad\quad \lambda x'.\, \lambda y.\, ((\lambda b.\, \lambda x''.\, \lambda y'.\, b\,y'\,x'')\,(\lambda z.\, \lambda w.\, z))\,y\,x' \Rightarrow \lambda x'.\, \lambda y.\, ((\lambda b.\, \lambda x''.\, \lambda y'.\, b\,y'\,x'')\,(\lambda z.\, \lambda w.\, z))\,y\,x' \\
\quad \Rightarrow \lambda x'.\, \lambda y.\, ((\lambda b.\, \lambda x''.\, \lambda y'.\, b\,y'\,x'')\,(\lambda z.\, \lambda w.\, z))\,y\,x' \\
\Rightarrow \lambda x'.\, \lambda y.\, ((\lambda b.\, \lambda x''.\, \lambda y'.\, b\,y'\,x'')\,(\lambda z.\, \lambda w.\, z))\,y\,x'
\end{array}
$$

1.4 Proofs of Eager Evaluations

We use the indented proof style.

$$
\begin{array}{l}
(\lambda f.\, \lambda x.\, f\,(f\,x))\,(\lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x)\,(\lambda z.\, \lambda w.\, z) \\
\quad (\lambda f.\, \lambda x.\, f\,(f\,x))\,(\lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x) \\
\quad\quad \lambda f.\, \lambda x.\, f\,(f\,x) \Rightarrow \lambda f.\, \lambda x.\, f\,(f\,x) \\
\quad\quad \lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x \Rightarrow \lambda b.\, \lambda x.\, \lambda y.\, b\,y\,x \\
\quad\quad (\lambda x.\, (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,x)) \Rightarrow (\lambda x.\, (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,x)) \\
\quad \Rightarrow (\lambda x.\, (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,x)) \\
\quad \lambda z.\, \lambda w.\, z \Rightarrow \lambda z.\, \lambda w.\, z \\
\quad (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,((\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,(\lambda z.\, \lambda w.\, z)) \\
\quad\quad \lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x' \Rightarrow \lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x' \\
\quad\quad (\lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x')\,(\lambda z.\, \lambda w.\, z) \\
\quad\quad\quad \lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x' \Rightarrow \lambda b.\, \lambda x'.\, \lambda y.\, b\,y\,x' \\
\quad\quad\quad \lambda z.\, \lambda w.\, z \Rightarrow \lambda z.\, \lambda w.\, z \\
\quad\quad\quad \lambda x'.\, \lambda y.\, (\lambda z.\, \lambda w.\, z)\,y\,x' \Rightarrow \lambda x'.\, \lambda y.\, (\lambda z.\, \lambda w.\, z)\,y\,x' \\
\quad\quad \Rightarrow \lambda x'.\, \lambda y.\, (\lambda z.\, \lambda w.\, z)\,y\,x' \\
\quad\quad \lambda x.\, \lambda y.\, (\lambda x'.\, \lambda y'.\, (\lambda z.\, \lambda w.\, z)\,y'\,x')\,y\,x) \Rightarrow \lambda x.\, \lambda y.\, (\lambda x'.\, \lambda y'.\, (\lambda z.\, \lambda w.\, z)\,y'\,x')\,y\,x) \\
\quad \Rightarrow \lambda x.\, \lambda y.\, (\lambda x'.\, \lambda y'.\, (\lambda z.\, \lambda w.\, z)\,y'\,x')\,y\,x) \\
\Rightarrow \lambda x.\, \lambda y.\, (\lambda x'.\, \lambda y'.\, (\lambda z.\, \lambda w.\, z)\,y'\,x')\,y\,x)
\end{array}
$$

2 Reynolds Problem 10.2

> Prove that, for any clased expression e and canonical form z, the expression e evaluates eagerly to z if and only if there is a proof of $e \Rightarrow_E z$ from the inference rules in Section 10.4.

2.1 Soundness

We first prove the *if* half of the proposition, which is that the rules for eager evaluation are sound.

If the last line of the proof is obtained by the rule for termination, it has the form $\lambda\mathsf{v}.\ e \Rightarrow \lambda\mathsf{v}.\ e$, which holds because $\lambda\mathsf{v}.\ e$ is a canonical form.

Otherwise, suppose the last line of the proof is obtained by the rule for beta-evaluation. Then it has the form $ee' \Rightarrow_E z$, and there are shorter proofs of the premisses $e \Rightarrow_E \lambda\mathsf{v}.\ \hat{e}$, $e' \Rightarrow_E z'$ and $(\hat{e}/v \rightarrow z') \Rightarrow_E z$, which by the induction hypothesis must be true. From the first premiss, there is an eager reduction sequence

$$e \rightarrow \cdots \rightarrow \lambda\mathsf{v}.\ \hat{e}$$

where the last term is the first canonical form. For the second premiss, there is an eager reduction sequence

$$e' \rightarrow \cdots \rightarrow z'$$

where the last term is the first canonical form. In other words, This implies that

$$ee' \rightarrow \cdots \rightarrow (\lambda\mathsf{v}.\ \hat{e})e' \rightarrow \cdots \rightarrow (\lambda\mathsf{v}.\ \hat{e})z'$$

is a reduction sequence where each term is contracted on the leftmost redex that's not a subexpression of a canonical form.

Now we can complete this sequence with a beta-contraction and the sequence whose existence is asserted by the third premiss:

$$ee' \rightarrow \cdots \rightarrow (\lambda\mathsf{v}.\ \hat{e})e' \rightarrow \cdots \rightarrow (\lambda\mathsf{v}.\ \hat{e})z' \rightarrow (\hat{e}/v \rightarrow z') \rightarrow \cdots \rightarrow z$$

This is an eager reduction sequence in which z is the first canonical form; thus $ee' \Rightarrow_E z$.

2.2 Completeness

Next, we prove the *only if* half of the proposition, which is that the eager evaluation rules are complete. Here we assume that we have a eager reduction sequence $e \rightarrow \cdots \rightarrow z$ in which only the last term is a canonical form, and we show, by induction on the length of the reduction sequence, the existence of the corresponding prooof of $e \Rightarrow_E z$.

If the reduction sequence begins with an abstraction, it must consist of the single term $\lambda\mathsf{v}.\ e$, whithout any constractions, and the rcorresponding proof is a single use of the rule for termination.

Otherwise, if the reduction sequence begins with an application, let $n \geq 0$ be the index of the first term in this reduction sequence that is not an application whose left subterm is also an application. (Such a term must exist since z does not have this form.) Then the reduction sequence has the form

$$e_0e'_0 \to \cdots \to e_{n-1}e'_{n-1} \to \ldots$$

where $e_0, \ldots, e_{n-1}$ are all applications, and therefore, by Proposition 10.5, contain redices. Moreover, the fact that this reduction sequence is eager implies that it has the form

$$e_0e'_0 \to \cdots \to e_{n-1}e'_0 \to e_ne'_0 \to \ldots$$

where

$$e_0 \to \cdots \to e_{n-1} \to e_n$$

is also an eager reduction sequence. This is because eager reduction always contracts the leftmost redex that's not a subexpression of a canonical form.

Further more, since e_ne' becomes an application, let $m \geq 0$ be the index of the second term in this reduction sequence that is an application whose right subterm is not an application. (Such a term must exist since z does not have this form.) Then the reduction sequence has the form

$$e_0e'_0 \to \cdots \to e_ne'_0 \to \cdots \to e_ne'_m$$

where

$$e'_0 \to \cdots \to e'_{m-1} \to e'_m$$

is also an eager reduction sequence. This is because e_n is already in canonical form, and eager evaluation may only contracts the right term after n steps.

Since $e_n = \lambda \mathsf{v}.\ \hat{e}$, and $e'_m = z'$, the term $e_ne'_m$ is a redex, and the rest of the original reduction sequence must have the form

$$(\lambda \mathsf{v}.\ \hat{e})z' \to (\hat{e}/v \to z') \cdots \to z$$

where only the last term is a canonical form. Then the induction hypothesis implies that there is a proof of $ee' \Rightarrow_E z$.

Finally, from the proofs of $e_0 \Rightarrow_E \lambda \mathsf{v}.\ \hat{e}, e'_0 \Rightarrow_E z'$, and $(\hat{e}/v \to z') \Rightarrow_E z$, one can use the rule for beta evaluation to construct a proof of $e_0e'_0 \Rightarrow_E z$.

3 Reynolds Problem 10.5

Given $((ab\ f)\ x)\ y = f <x, y>$, and a continuous function f from $P \times P'$ to P''

1. Prove $(ab\ f)\ x$ is a continuous function from P' to P''.
2. Prove $ab\ f$ is a continuation function from P to $P' \to P''$

3.1

We can rewrite the function $(ab\ f)\ x$ as

$$
\begin{aligned}
&(ab\ f)\ x = f \cdot g \\
&\qquad \text{where } g\ y = <x, y>
\end{aligned}
$$

We'll first show that g is a continuous function, and then by by Proposition 2.3 Part (c), $(ab\ f)x$ is continuous.

3.1.1 g is monotone

For any $y_1 \sqsubseteq y_2 \in P'$, we have $g\ y_1 = <x, y_1> \sqsubseteq <x, y_2> = g\ y_2$ due to the ordering on the product of pre-domain $P \times P'$.

3.1.2 g is continuous

For any interesting chain $y_i \in P'$, we have

$$g(\bigsqcup_{i=0}^{\infty}{}' y_i) = <x, \bigsqcup_{i=0}^{\infty}{}' y_i> = \bigsqcup_{i=0}^{\infty}{}^{\times} <x, y_i> = \bigsqcup_{i=0}^{\infty}{}^{\times} (g\ y_i)$$

Therefore by Proposition 2.3 Part (c), $(ab\ f)\ x = f \cdot g$ is a continuous function from P' to P''.

3.2

3.2.1 *ab f* is monotone

For any $x_1 \sqsubseteq x_2 \in P$, let $h_1 = (ab\ f)\ x_1$ and $h_2 = (ab\ f)\ x_2$. We want to show $h_1 \sqsubseteq h_2$, i.e., forall $y \in P'$, $h_1\ y \sqsubseteq h_2\ y$.

This is true because $h_1\ y = f\ <x_1, y> \sqsubseteq f\ <x_2, y> = h_2\ y$ due to the ordering on the product of pre-domain $P \times P'$, and the continuity of f. Therefore $(ab\ f)\ x_1 \sqsubseteq (ab\ f)\ x_2$ and $ab\ f$ is monotone.

3.2.2 *ab f* is continuous

for any interesting chain $x_i \in P$, and any $y \in P'$, then $F_i = (ab\ f)x_i$ also forms a chain of functions in the pre-domain $P' \to P''$.

Because we have proved that $((ab\ f)x)$ is continuous, by Proposition 2.2 we have:

$$(\bigsqcup_{i=0}^{\infty}{}^{\rightarrow} F_i)y = \bigsqcup_{i=0}^{\infty}{}''(F_i\ y)$$

Or in another word

$$(\bigsqcup_{i=0}^{\infty}{}^{\rightarrow}((ab\ f)x_i))y = \bigsqcup_{i=0}^{\infty}{}''(((ab\ f)x_i)y) = \bigsqcup_{i=0}^{\infty}{}''(f <x_i, y>) = f(\bigsqcup_{i=0}^{\infty}{}^{\times} <x_i, y>)$$

because f is continuous. Furthermore

$$f(\bigsqcup_{i=0}^{\infty}{}^{\times} <x_i, y>) = f <\bigsqcup_{i=0}^{\infty} x_i, y> = (ab\ f)(\bigsqcup_{i=0}^{\infty} x_i)y$$

So we have effective proved that for any $y \in P'$

$$(ab\ f)(\bigsqcup_{i=0}^{\infty} x_i)y = (\bigsqcup_{i=0}^{\infty}{}^{\rightarrow}((ab\ f)x_i))y$$

Therefore

$$(ab\ f)(\bigsqcup_{i=0}^{\infty} x_i) = \bigsqcup_{i=0}^{\infty}{}^{\rightarrow}((ab\ f)x_i)$$

and $ab\ f$ is continuous.