CMPT 383 Comparative Programming Languages

Homework 5 Solution

This homework is due by 11:59pm PT on Wednesday Mar 16, 2022. No late submission is accepted. Please save your answers in a single file called `h5_firstname_lastname.pdf` and submit it to Canvas. You may also write on paper and scan it (or take a picture) into a PDF. Please make sure the text is readable.

Requirements of this homework:

- Use normal order strategy for beta reductions.

1. (20 points) Reduce the following λ-term to normal form

$$(\lambda x.\ x\ y\ x)\ (\lambda z.z)$$

Solution:

$$\begin{array}{ll} & (\lambda x.\ x\ y\ x)\ (\lambda z.z) \\ \to & (\lambda z.z)\ y\ (\lambda z.z) \\ \to & y\ (\lambda z.z) \end{array}$$

2. (20 points) Given Church numerals and $times = \lambda m.\lambda n.\lambda s.\lambda z.\ m\ (n\ s)\ z$, show your steps to check

$$times\ 1\ 2 \to^* 2$$

Solution:

$$\begin{array}{ll} & times\ 1\ 2 \\ = & (\lambda m.\lambda n.\lambda s.\lambda z.\ m\ (n\ s)\ z)\ 1\ 2 \\ \to & (\lambda n.\lambda s.\lambda z.\ 1\ (n\ s)\ z)\ 2 \\ \to & \lambda s.\lambda z.\ 1\ (2\ s)\ z \\ = & \lambda s.\lambda z.\ (\lambda s'.\lambda z'.\ s'\ z')\ (2\ s)\ z \\ \to & \lambda s.\lambda z.\ (\lambda z'.\ (2\ s)\ z')\ z \\ \to & \lambda s.\lambda z.\ (2\ s)\ z \\ = & \lambda s.\lambda z.\ ((\lambda s'.\lambda z'.\ s'\ (s'\ z'))\ s)\ z \\ \to & \lambda s.\lambda z.\ (\lambda z'.\ s\ (s\ z'))\ z \\ \to & \lambda s.\lambda z.\ s\ (s\ z) \\ = & 2 \end{array}$$

The bound variables s' and z' do not necessarily need to be primed or renamed.

3. (20 points) Given the following definition of Y combinator:

$$Y = \lambda f.\ (\lambda x.\ f\ (x\ x))\ (\lambda x.\ f\ (x\ x))$$

Check $Y\ g = g\ (Y\ g)$ by showing that Yg and $g\ (Y\ g)$ reduce to the same λ-term.

Solution:

$$\begin{array}{ll} & Y\ g \\ = & (\lambda f.\ (\lambda x.\ f\ (x\ x))\ (\lambda x.\ f\ (x\ x)))\ g \\ \to & (\lambda x.\ g\ (x\ x))\ (\lambda x.\ g\ (x\ x)) \\ \to & g\ ((\lambda x.\ g\ (x\ x))\ (\lambda x.\ g\ (x\ x))) \end{array}$$

$$
\begin{array}{rl}
& g\ (Y\ g) \\
= & g\ ((\lambda f.\ (\lambda x.\ f\ (x\ x))\ (\lambda x.\ f\ (x\ x)))\ g) \\
\rightarrow & g\ ((\lambda x.\ g\ (x\ x))\ (\lambda x.\ g\ (x\ x)))
\end{array}
$$

4. (20 points) Given the following definitions:

$$
\begin{array}{rcl}
S & = & \lambda f.\lambda g.\lambda x.\ f\ x\ (g\ x) \\
K & = & \lambda x.\lambda y.\ x
\end{array}
$$

Reduce $S\ K\ K$ to normal form.

Solution:

$$
\begin{array}{rl}
& S\ K\ K \\
= & (\lambda f.\lambda g.\lambda x.\ f\ x\ (g\ x))\ K\ K \\
\rightarrow & (\lambda g.\lambda x.\ K\ x\ (g\ x))\ K \\
\rightarrow & \lambda x.\ K\ x\ (K\ x) \\
= & \lambda x.\ (\lambda x.\lambda y.\ x)\ x\ (K\ x) \\
\rightarrow & \lambda x.\ (\lambda y.\ x)\ (K\ x) \\
\rightarrow & \lambda x.x
\end{array}
$$

5. (20 points) Define a λ-term called *or* such that *or* is a binary logical or function for Church booleans. Also, show your steps to check

$$
or\ false\ true \rightarrow^* true
$$

Solution:

$$
or = \lambda x.\lambda y.\ x\ true\ y
$$

$$
\begin{array}{rl}
& or\ true\ false \\
= & (\lambda x.\lambda y.\ x\ true\ y)\ false\ true \\
\rightarrow & (\lambda y.\ false\ true\ y)\ true \\
\rightarrow & false\ true\ true \\
= & (\lambda t.\lambda f.f)\ true\ true \\
\rightarrow & (\lambda f.f)\ true \\
\rightarrow & true
\end{array}
$$

Note that there are several correct definitions for *or*. Another definition is $or = \lambda x.\lambda y.\ x\ x\ y$