Semantics of Programming Languages:
Solution of Assignment 1

Thorsten Brunklaus and Jan Schwinghammer

Exercise 1.1: (5)

$$x \wedge y \overset{(def)}{=} \texttt{if x then y else x}$$
$$x \vee y \overset{(def)}{=} \texttt{if x then x else y}$$

Exercise 1.2: (5) $\lambda f.\, \lambda n.\, (\lambda x.\, x + x)\ (f\ n)$

Exercise 1.3: (5)

$$[(y+3)/x]\ ((\lambda f.\, \lambda y.\ f\ (x+y))\ (\lambda x.\, x+y)) = ((\lambda f.\, \lambda z.\ f\ ((y+3)+z))\ (\lambda x.\, x+y))$$

Exercise 1.4: (5+5)

(a)

$$\begin{aligned} p &\overset{(sp)}{=} \langle Proj_1\ p, Proj_2\ p \rangle \\ &\overset{(ass)}{=} \langle Proj_1\ q, Proj_2\ q \rangle \\ &\overset{(sp)}{=} q \end{aligned}$$

(b)

$$\begin{aligned} M &\overset{(\eta)}{=} \lambda x.\, M\ x \\ &\overset{(Ass)}{=} \lambda x.\, N\ x \\ &\overset{(\eta)}{=} N \end{aligned}$$

Exercise 1.5: (40)

$$Curry \overset{(def)}{=} \lambda f.\, \lambda x.\, \lambda y.\ f\ \langle x, y \rangle$$
$$Uncurry \overset{(def)}{=} \lambda g.\, \lambda x.\, g\ (Proj_1\ x)\ (Proj_2\ x)$$

(a)

$$
\begin{aligned}
Uncurry\ (Curry\ g) &\overset{(def)}{=} Uncurry\ ((\lambda f.\ \lambda x.\ \lambda y.\ f\ \langle x, y\rangle)\ g) \\
&\overset{(\beta)}{=} Uncurry\ (\lambda x.\ \lambda y.\ g\ \langle x, y\rangle) \\
&\overset{(def)}{=} (\lambda g.\ \lambda x.\ g\ (Proj_1\ x)\ (Proj_2\ x))\ (\lambda x.\ \lambda y.\ g\ \langle x, y\rangle) \\
&\overset{(\beta)}{=} \lambda p.\ (\lambda x.\ \lambda y.\ g\ \langle x, y\rangle)\ (Proj_1\ p)\ (Proj_2\ p) \\
&\overset{(\beta)}{=} \lambda p.\ (\lambda y.\ g\ \langle Proj_1\ p, y\rangle)\ (Proj_2\ p) \\
&\overset{(\beta)}{=} \lambda p.\ g\ \langle Proj_1\ p, Proj_2\ p\rangle \\
&\overset{(sp)}{=} \lambda p.\ g\ p \\
&\overset{(\eta)}{=} g
\end{aligned}
$$

(b)

$$
\begin{aligned}
Curry\ (Uncurry\ f) &\overset{(def)}{=} Curry\ ((\lambda g.\ \lambda x.\ g\ (Proj_1\ x)\ (Proj_2\ x))\ \ f) \\
&\overset{(\beta)}{=} Curry\ (\lambda x.\ f\ (Proj_1\ x)\ (Proj_2\ x)) \\
&\overset{(def)}{=} (\lambda f.\ \lambda x.\ \lambda y.\ f\ \langle x, y\rangle)\ (\lambda x.\ f\ (Proj_1\ x)\ (Proj_2\ x)) \\
&\overset{(\beta)}{=} \lambda x.\ \lambda y.\ (\lambda z.\ f\ (Proj_1\ z)\ (Proj_2\ z))\ \langle x, y\rangle \\
&\overset{(\beta)}{=} \lambda x.\ \lambda y.\ f\ (Proj_1\ \langle x, y\rangle)\ (Proj_2\ \langle x, y\rangle) \\
&\overset{(def)}{=} \lambda x.\ \lambda y.\ (f\ x)\ y \\
&\overset{(\eta)}{=} \lambda x.\ f\ x \\
&\overset{(\eta)}{=} f
\end{aligned}
$$

Exercise 1.6: (5)

$$
\begin{aligned}
\texttt{let x} = \texttt{M in N} &\overset{(def)}{=} (\lambda x.\ N)\ M \\
&\overset{(\beta)}{=} [N/x]\ M
\end{aligned}
$$

Exercise 1.7: (10)

(a)

$$\lambda\langle x, y\rangle.\ M \overset{(def)}{=} \lambda z.\ (\lambda x.\ \lambda y.\ M)\ (Proj_1\ z)\ (Proj_2\ z)$$

(b)

$$
\begin{aligned}
\texttt{let f}\langle \texttt{x, y}\rangle = \texttt{M in N} &\overset{(def)}{=} \texttt{let f} = \lambda \texttt{z.}\ (\lambda \texttt{x.}\ \lambda \texttt{y.}\ \texttt{M})\ (\texttt{Proj}_1\ \texttt{z})\ (\texttt{Proj}_2\ \texttt{z})\ \texttt{in N} \\
&\overset{(let)}{=} (\lambda f.\ N)\ (\lambda z.\ (\lambda x.\ \lambda y.\ M)\ (Proj_1\ z)\ (Proj_2\ z)) \\
&\overset{(\beta)}{=} [(\lambda z.\ (\lambda x.\ \lambda y.\ M)\ (Proj_1\ z)\ (Proj_2\ z))/f]\ N
\end{aligned}
$$

Exercise 1.8: (15)

$$R \stackrel{def}{=} \texttt{if Eq? x 0 then 1 else if Eq? x 1 then 1 else fib (x - 1) + fib (x - 2)}$$

$$F \stackrel{def}{=} \lambda fib.\, \lambda x.\, R$$

$$\begin{aligned}
\texttt{letrec fib(x) = R in fib 4} &\to^* \texttt{let fib = fix F in fib 4} \\
&\to^* (\lambda fib.\ fib\ 4)\ (\texttt{fix F}) \\
&\to^* (\texttt{fix F})\ 4 \\
&\to^* (F\ (\texttt{fix F}))\ 4 \\
&\to^* (\lambda x.\, [(\texttt{fix F})/\texttt{fib}]\ \texttt{R})\ 4 \\
&\to^* \texttt{if Eq? 4 0 then 1 else if Eq? 4 1 then 1 else (fix F) (4 - 1) + (fix F) (4 - 2)} \quad \to^* 5
\end{aligned}$$

Exercise 1.9: (5)

(a)

$$\begin{aligned}
M &\stackrel{(cond)}{=} \texttt{if true then M else N} \\
&\stackrel{(Ass)}{=} \texttt{if false then M else N} \\
&\stackrel{(cond)}{=} N
\end{aligned}$$

(b)

$$\begin{aligned}
M &\stackrel{(cond)}{=} \texttt{if true then M else N} \\
&\stackrel{(=)}{=} \texttt{if (n = n) then M else N} \\
&\stackrel{(Ass)}{=} \texttt{if (n = m) then M else N} \\
&\stackrel{(=)}{=} \texttt{if false then M else N} \\
&\stackrel{(cond)}{=} N
\end{aligned}$$