Elements of Programming Languages
Tutorial 2: Substitution and alpha-equivalence
Solution notes

1. (a)
   - $(\lambda x{:}\texttt{int}.\ x)\ 1$

$$\dfrac{\dfrac{}{\lambda x{:}\texttt{int}.\ x \Downarrow \lambda x{:}\texttt{int}.\ x} \quad \dfrac{}{1 \Downarrow 1} \quad \dfrac{}{1 \Downarrow 1}}{(\lambda x{:}\texttt{int}.\ x)\ 1 \Downarrow 1}$$

   - $(\lambda x{:}\texttt{int}.\ x+1)\ 42$

$$\dfrac{\dfrac{}{\lambda x{:}\texttt{int}.\ x+1 \Downarrow \lambda x{:}\texttt{int}.\ x+1} \quad \dfrac{}{42 \Downarrow 42} \quad \dfrac{\dfrac{}{42 \Downarrow 42} \quad \dfrac{}{1 \Downarrow 1}}{42+1 \Downarrow 43}}{(\lambda x{:}\texttt{int}.\ x+1)\ 42 \Downarrow 43}$$

   - $(\lambda x{:}\texttt{int} \to \texttt{int}.\ x)\ (\lambda x{:}\texttt{int}.\ x)\ 1$ Type annotations elided.

$$\dfrac{\dfrac{\dfrac{}{\lambda x.\ x \Downarrow \lambda x.\ x} \quad \dfrac{}{\lambda x.\ x \Downarrow \lambda x.\ x} \quad \dfrac{}{\lambda x.\ x \Downarrow \lambda x.\ x}}{(\lambda x.\ x)\ (\lambda x.\ x) \Downarrow \lambda x.\ x} \quad \dfrac{}{1 \Downarrow 1}}{((\lambda x.\ x)\ (\lambda x.\ x))\ 1 \Downarrow 1}$$

   - $(\star)$ $((\lambda f{:}\texttt{int} \to \texttt{int}.\ \lambda x{:}\texttt{int}.f\ (f\ x))\ (\lambda x{:}\texttt{int}.\ x+1))\ 42$ Type annotations elided.

$$\dfrac{\dfrac{}{(\lambda f.\ \lambda x.f\ (f\ x))\ (\lambda x.\ x+1) \Downarrow \lambda x.(\lambda x.\ x+1)((\lambda x.\ x+1)x)} \quad \dfrac{}{42 \Downarrow 42} \quad \dfrac{\vdots}{(\lambda x.\ x+1)((\lambda x.\ x+1)42) \Downarrow 44}}{((\lambda f.\ \lambda x.f\ (f\ x))\ (\lambda x.\ x+1))\ 42 \Downarrow 44}$$

   where

$$\dfrac{\dfrac{}{\lambda x.\ x+1 \Downarrow \lambda x.\ x+1} \quad \dfrac{\dfrac{}{\lambda x.\ x+1 \Downarrow \lambda x.x+1} \quad \dfrac{}{42 \Downarrow 42} \quad \dfrac{}{42+1 \Downarrow 43}}{(\lambda x.\ x+1)42 \Downarrow 43} \quad \dfrac{}{43+1 \Downarrow 44}}{(\lambda x.\ x+1)((\lambda x.\ x+1)42) \Downarrow 44}$$

   (b) If $e_1 : \tau$ then we can define $\texttt{let}\ x = e_1\ \texttt{in}\ e_2$ as $(\lambda x{:}\tau.\ e_2)\ e_1$. The evaluation rule for `let` can be emulated as follows:

$$\dfrac{e_1 \Downarrow v_1 \quad e_2[v_1/x] \Downarrow v}{\texttt{let}\ x = e_1\ \texttt{in}\ e_2 \Downarrow v} \implies \dfrac{\dfrac{}{\lambda x{:}\tau.e_2 \Downarrow \lambda x{:}\tau.\ e_2} \quad e_1 \Downarrow v_1 \quad e_2[v_1/x] \Downarrow v}{(\lambda x{:}\tau.\ e_2)\ e_1 \Downarrow v}$$

2. (a)
   - `Int => Int`

```
{x: Int => x}
```

   - `Int => Boolean => Int`

```
{x: Int => {y: Boolean => x}}
```

   - `(Int => Boolean => String) => (Int => Boolean) => (Int => String)`

```
{x: (Int => Boolean => String) =>
  {y: (Int => Boolean) =>
    {z: Int => x(z)(y(z))}}}
```

(b)
- $(\lambda x{:}\mathtt{int}.\ x)\ 1$

$$\dfrac{\dfrac{\overline{x : \mathtt{int} \vdash x : \mathtt{int}}}{\vdash \lambda x{:}\mathtt{int}.\ x : \mathtt{int} \to \mathtt{int}} \qquad \overline{\vdash 1 : \mathtt{int}}}{\vdash (\lambda x{:}\mathtt{int}.\ x)\ 1 : \mathtt{int}}$$

- $(\lambda x{:}\mathtt{int}.\ x + 1)\ 42$

$$\dfrac{\dfrac{\dfrac{\overline{x{:}\mathtt{int} \vdash x : \mathtt{int}} \quad \overline{x{:}\mathtt{int} \vdash 1 : \mathtt{int}}}{x : \mathtt{int} \vdash x + 1 : \mathtt{int}}}{\vdash \lambda x{:}\mathtt{int}.\ x + 1 : \mathtt{int} \to \mathtt{int}} \qquad \overline{\vdash 42 : \mathtt{int}}}{\vdash (\lambda x{:}\mathtt{int}.\ x + 1)\ 42 : \mathtt{int}}$$

- $(\lambda x{:}\mathtt{int} \to \mathtt{int}.\ x)\ (\lambda x{:}\mathtt{int}.\ x)$

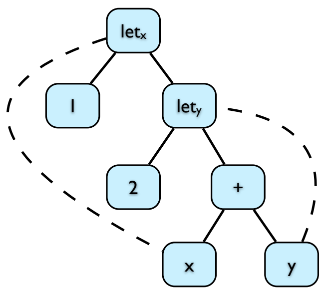
$$\dfrac{\dfrac{\overline{x{:}\mathtt{int} \to \mathtt{int} \vdash x : \mathtt{int} \to \mathtt{int}}}{\vdash (\lambda x{:}\mathtt{int} \to \mathtt{int}.\ x) : (\mathtt{int} \to \mathtt{int}) \to (\mathtt{int} \to \mathtt{int})} \qquad \dfrac{\vdots}{\vdash \lambda x{:}\mathtt{int}\ x : \mathtt{int} \to \mathtt{int}}}{\vdash (\lambda x{:}\mathtt{int} \to \mathtt{int}.\ x)\ (\lambda x{:}\mathtt{int}.\ x) : \mathtt{int} \to \mathtt{int}}$$

- $(\lambda x{:}\tau.\ x\ x)$ **This expression cannot be typed. There is no way to choose τ so that the following derivation can be completed:**

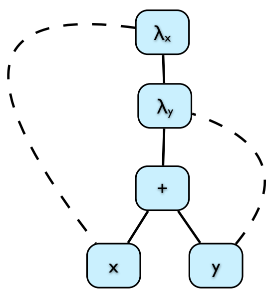
$$\dfrac{\dfrac{\dfrac{??}{x{:}\tau \vdash x : \tau_1 \to \tau_2} \quad \dfrac{??}{x{:}\tau \vdash x{:}\tau_1}}{x{:}\tau \vdash x\ x : \tau_2}}{\vdash \lambda x{:}\tau.\ x\ x : \tau_2}$$

For if $\tau = \tau_1$ then we would also have to have $\tau = \tau_1 \to \tau_2$, i.e. $\tau_1 = \tau_1 \to \tau_2$ which is not possible if equality is structural.

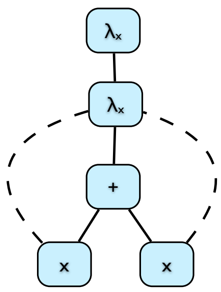
3. (a) The pictures should be as follows:



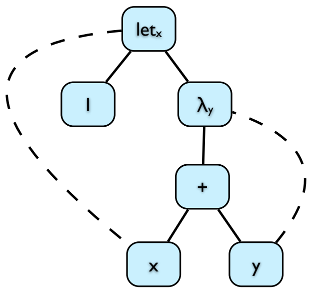
let x = 1 in let y = 2 in x + y



λ x. λ y. x + y



λ x. λ x. x + x



let x = 1 in λ y. x + y

(b) The missing rules are:

$\boxed{e_1 \equiv_\alpha e_2}$

$$\frac{e \equiv_\alpha e' \quad e_1 \equiv_\alpha e_1' \quad e_1 \equiv_\alpha e_1'}{\texttt{if } e \texttt{ then } e_1 \texttt{ else } e_2 \equiv_\alpha \texttt{if } e' \texttt{ then } e_1' \texttt{ else } e_2'}$$

$$\frac{e_1[z/x] \equiv_\alpha e_2[z/y] \quad z \notin FV(e_1) cup FV(e_2)}{\lambda x.e_1 \equiv_\alpha \lambda y.e_2} \qquad \frac{e_1 \equiv_\alpha e_1' \quad e_1 \equiv_\alpha e_1'}{e_1\ e_2 \equiv_\alpha e_1'\ e_2'}$$

Point this out: To be precise, we should also extend FV as follows:

$$\begin{aligned} FV(\texttt{if } e \texttt{ then } e_1 \texttt{ else } e_2) &= FV(e) \cup FV(e_1) \cup FV(e_2) \\ FV(\lambda x{:}\tau.\ e) &= FV(e) - \{x\} \\ FV(e_1) \cup FV(e_2) &= FV(e_1) \cup FV(e_2) \end{aligned}$$

(c) Which of the following alpha-equivalence relationships hold?

$\texttt{if true then } y \texttt{ else } z \equiv_\alpha y$	FALSE
$\texttt{let } x = y \texttt{ in } (\texttt{if } x \texttt{ then } y \texttt{ else } z) \equiv_\alpha \texttt{let } z = y \texttt{ in } (\texttt{if } x \texttt{ then } y \texttt{ else } z)$	FALSE
$\texttt{let } x = 1 \texttt{ in } (\texttt{let } y = x \texttt{ in } y + y) \equiv_\alpha \texttt{let } x = 1 \texttt{ in } (\texttt{let } x = x \texttt{ in } x + x)$	TRUE
$\lambda x.\lambda y.x\ y \equiv_\alpha \lambda y.\lambda x.y\ x$	TRUE
$\lambda y.x\ y \equiv_\alpha \lambda x.y\ x$	FALSE

4. (a) The missing cases are:

$$\begin{aligned} (\texttt{if } e_0 \texttt{ then } e_1 \texttt{ else } e_2)[e/x] &= \texttt{if } e_0[e/x] \texttt{ then } e_1[e/x] \texttt{ else } e_2[e/x] \\ (e_1\ e_2)[e/x] &= e_1[e/x]\ e_2[e/x] \\ (\lambda x{:}\tau.e_0)[e/x] &= \lambda x{:}\tau.e_0 \\ (\lambda y{:}\tau.e_0)[e/x] &= \lambda y{:}\tau.e_0[e/x] \qquad (y \notin FV(e)) \end{aligned}$$

Make sure it is clear why the side condition is necessary for λ.

(b)

$$\begin{aligned} (\lambda y.\ \lambda z.\ ((x+y)+z))[y \times z/x] &\equiv_\alpha (\lambda y'.\ \lambda z'.\ ((x+y')+z'))[y \times z/x] \\ &= \lambda y'.\ \lambda z'.\ ((y \times z + y') + z') \end{aligned}$$

$$\begin{aligned} (\texttt{if } x == y \texttt{ then } \lambda z.x \texttt{ else } \lambda x.x)[z/x] &\equiv_\alpha (\texttt{if } x == y \texttt{ then } \lambda z'.x \texttt{ else } \lambda x'.x')[z/x] \\ &= \texttt{if } z == y \texttt{ then } \lambda z'.z \texttt{ else } \lambda x'.x' \end{aligned}$$