(a)

1 Axiom (var≻): $\Gamma \vdash x : \tau$ if $(x:\sigma) \in \Gamma$ and $\sigma \succ \tau$

1 where $\sigma \succ \tau$ (σ generalises τ) means $\sigma = \forall \alpha_1 \ldots \alpha_n (\tau')$ say, and τ is obtained from τ' by substituting some types: $\tau = \tau'[\tau_1/\alpha_1, \ldots, \tau_n/\alpha_n]$.

2 Rule (fn): $$\frac{\Gamma, x : \forall \emptyset (\tau_1) \vdash M : \tau_2}{\Gamma \vdash \lambda x (M) : \tau_1 \to \tau_2} \text{ if } x \notin dom(\Gamma)$$

1 Rule (app): $$\frac{\Gamma \vdash M_1 : \tau_1 \to \tau_2 \quad \Gamma \vdash M_2 : \tau_1}{\Gamma \vdash M_1 M_2 : \tau_2}$$

2 Rule (let): $$\frac{\Gamma \vdash M_1 : \tau_1 \qquad \Gamma, x : \forall A (\tau_1) \vdash M_2 : \tau_2}{\Gamma \vdash \text{let } x = M_1 \text{ in } M_2 : \tau_2} \left\{ \begin{array}{l} \text{if } x \notin dom(\Gamma) \\ \text{and } A = \\ ftv(\tau_1) - ftv(\Gamma) \end{array} \right.$$

7

(b)

1 There is no τ for which $\emptyset \vdash Y : \tau$ holds, because any proof of such a

1 typing would have to look like

$$\Phi \vdash Y : \tau$$

(fn)

$$x : \tau_1 \vdash (\lambda y(x(yy)))\lambda y(x(yy)) : \tau_2$$

(app)

$$x : \tau_1 \vdash \lambda y(x(yy)) : \tau_3 \qquad x : \tau_1 \vdash \lambda y(x(yy)) : ?$$

(fn)

$$\underbrace{x : \tau_1, y : \tau_4}_{\Gamma} \vdash x(yy) : \tau_5$$

(app)

$$\Gamma \vdash x : \tau_6 \qquad \Gamma \vdash yy : \tau_7$$

(var) ③ (app)

2 $$\Gamma \vdash y : \tau_8 \qquad \Gamma \vdash y : \tau_9$$

① (var) ② (var)

for some $\tau_1, \tau_2, \ldots, \tau_9$ with

① $\tau_8 = \tau_4$

② $\tau_9 = \tau_4$

③ $\tau_8 = \tau_9 \to \tau_7$

& hence $\tau_4 \overset{①}{=} \tau_8 \overset{③}{=} \tau_9 \to \tau_7 \overset{②}{=} \tau_4 \to \tau_7$,

2 but no such τ_4 & τ_7 can exist, by counting

6 # of $\to$ symbols in each.

(c) Let $\Gamma = \{x : \omega \to \alpha, y : \omega \to \omega\}$. Then:

$$\dfrac{\dfrac{}{\Gamma \vdash x : \omega \to \alpha}\,(\text{var}) \qquad \dfrac{\dfrac{}{\Gamma \vdash y : \omega \to \omega}\,(\text{var}) \quad \dfrac{}{\Gamma \vdash y : \omega}\,(\text{univ})}{\Gamma \vdash yy : \omega}\,(\text{app})}{\dfrac{\Gamma \vdash x(yy) : \alpha}{x : \omega \to \alpha \vdash \lambda y(x(yy)) : (\omega \to \omega) \to \alpha}\,(\text{fn})}\,(\text{app})$$

3 as required.

We also have

$$\dfrac{\dfrac{}{x:\omega\to\alpha, y:\omega \vdash x(yy):\omega}\,(univ)}{x:\omega\to\alpha \vdash \lambda y(x(yy)):\omega\to\omega}\,(fn)$$

2

Then (app) on $\begin{cases} x:\omega\to\alpha \vdash \lambda y(x(yy)):(\omega\to\omega)\to\alpha \\ x:\omega\to\alpha \vdash \lambda y(x(yy)):\omega\to\omega \end{cases}$

gives

2 $x:\omega\to\alpha \vdash (\lambda y(x(yy)))\,\lambda y(x(yy)):\alpha$

& hence by (fn) $\phi \vdash Y:(\omega\to\alpha)\to\alpha$.

4

Commentary

Part (a) is bookwork from lectures 2-4.
Part (b) is a simple calculation, ~~typical~~ similar to ones done in lectures 2-4.
Part (c) is a new problem: the first part is relatively easy & helps to solve the last part. The expectation is that even though Y is a complicated-looking λ-term, candidates will have met it in the Foundations of Functional Programming course last year.