Introduction to Computational Logic, SS 2006: Solution for assignment 2

Prof. Dr. Gert Smolka, Dipl.-Inform. Mathias Möhl

Exercise 2.1 (Schönfinkel's U) Idea: Find terms, whose simplification (via $\beta\eta$-reduction and logical equivalence) gives the term we are after.

a) $\neg = \lambda b \in \mathbb{B}.U(\lambda x \in X.b)(\lambda x \in X.b)$

b) $\vee = \lambda b \in \mathbb{B}.\lambda c \in \mathbb{B}.(U(\lambda x \in X.\neg b)(\lambda x \in X.\neg c))$

c) $\exists = \lambda f \in X \to \mathbb{B}.\neg(Uff)$

Exercise 2.2 (Henkin's Reduction)

a) $1 = (\lambda x \in \mathbb{B}.x) \doteq_{\mathbb{B}\to\mathbb{B}} (\lambda x \in \mathbb{B}.x)$

b) $0 = (\lambda x \in \mathbb{B}.x) \doteq_{\mathbb{B}\to\mathbb{B}} (\lambda x \in \mathbb{B}.1)$

c) $\neg = \lambda x \in \mathbb{B}.x \doteq_{\mathbb{B}} 0$

d) $\wedge = \lambda x \in \mathbb{B}.\lambda y \in \mathbb{B}.(\lambda f \in \mathbb{B} \to \mathbb{B} \to \mathbb{B}.fxy) \doteq_{(\mathbb{B}\to\mathbb{B}\to\mathbb{B})\to\mathbb{B}} (\lambda f \in \mathbb{B} \to \mathbb{B} \to \mathbb{B}.f11)$

e) $\forall_X = \lambda f \in X \to \mathbb{B}.f \doteq_{X\to\mathbb{B}} (\lambda x \in X.1)$

Exercise 2.3 (Choice Function)

a) Idea: 0 is the only number in $\mathbb{N}$, which does not change when added to itself.
$0 = \mathbf{C}(\lambda x \in \mathbb{N}.x + x \doteq x)$

b) Idea: 1 and 0 are the only numbers in $\mathbb{N}$ that do not change when multiplied with themselves.
$1 = \mathbf{C}(\lambda x \in \mathbb{N}.x \cdot x \doteq x \wedge \neg(x + x \doteq x))$

c) $- = \lambda x \in \mathbb{N}.\lambda y \in \mathbb{N}.\mathbf{C}(\lambda z \in \mathbb{N}.x \doteq y + z)$

d) $\mathit{div} = \lambda x \in \mathbb{N}.\lambda y \in \mathbb{N}.\mathbf{C}(\lambda z \in \mathbb{N}.(0 \le x - z \cdot y) \wedge (x - z \cdot y \le y - 1))$

e) $\mathit{max} = \lambda x \in \mathbb{N}.\lambda y \in \mathbb{N}.\mathbf{C}(\lambda z \in \mathbb{N}.x \le z \wedge y \le z \wedge (z \le x \vee z \le y))$

f) Idea: Construct a set, which depending on b contains (only) x or y. The choice operator selects then exactly x resp. y.
$\lambda b \in \mathbb{B}.\lambda x \in \mathbb{N}.\lambda y \in \mathbb{N}.\mathbf{C}(\lambda z \in \mathbb{N}.(b \wedge (z \doteq x)) \vee (\neg b \wedge (z \doteq y)))$

g) Idea: It directly follows from the definition of $\mathbf{C}$. (Or another idea, if you interpret f as a set: Only if $f = \emptyset$, $\mathbf{C}f$ is not an element of f and simultaneously only if $f = \emptyset$, $\exists f$ does not hold.)
$\exists = \lambda f \in \mathbb{N} \to \mathbb{B}.f(\mathbf{C}f)$

Exercise 2.4 (Normal Forms)

a) $\lambda xy.fx$ is already in $\beta\eta$-normal form.

b)

$$\lambda xy.fy \equiv \lambda x.f \qquad (\eta)$$

c)

$$\begin{aligned} \lambda xy.fxy &\equiv \lambda x.fx && (\eta) \\ &\equiv f && (\eta) \end{aligned}$$

d)

$$\begin{aligned} (\lambda xy.fyx)zxy &\equiv (\lambda y.fyz)xy && (\beta) \\ &\equiv fxzy && (\beta) \end{aligned}$$

e)

$$\begin{aligned} (\lambda xy.fxy)ab &\equiv (\lambda x.fx)ab && (\eta) \\ &\equiv fab && (\eta) \end{aligned}$$

or alternatively:

$$\begin{aligned} (\lambda xy.fxy)ab &\equiv (\lambda y.fay)b && (\beta) \\ &\equiv fab && (\beta) \end{aligned}$$

f)

$$(\lambda z.a)(\lambda xy.fxy) \equiv a \qquad (\beta)$$

g)

$$(\lambda xu.x)(\lambda vy.y) \equiv \lambda uvy.y \qquad (\beta)$$

h)

$$\begin{aligned} (\lambda f.fa)(\lambda xy.fxy)b &\equiv (\lambda xy.fxy)ab && (\beta) \\ &\equiv fab && (\text{Exercise 2.4 e)}) \end{aligned}$$

i)

$$
\begin{aligned}
(\lambda z.(\lambda xy.fxy)((\lambda z.z)z))ab &\equiv (\lambda z.(\lambda xy.fxy)z)ab && (\beta)\\
&\equiv (\lambda xy.fxy)ab && (\eta)\\
&\equiv fab && \text{(Exercise 2.4 e))}
\end{aligned}
$$

j)

$$
\begin{aligned}
(\lambda fgx.fx(gx))(\lambda xy.x)(\lambda xz.x) &\equiv (\lambda gx.(\lambda xy.x)x(gx))(\lambda xz.x) && (\beta)\\
&\equiv (\lambda gx.(\lambda y.x)(gx))(\lambda xz.x) && (\beta)\\
&\equiv (\lambda gx.x)(\lambda xz.x) && (\beta)\\
&\equiv \lambda x.x && (\beta)
\end{aligned}
$$

Exercise 2.5 (Lambda Elimination)

a)

$$
\begin{aligned}
\lambda xy.y &\equiv K(\lambda y.y) && \text{(K)}\\
&\equiv KI && \text{(I)}
\end{aligned}
$$

b)

$$
\begin{aligned}
\lambda xyz.y &\equiv K(\lambda yz.y) && \text{(K)}\\
&\equiv K(\lambda y.Ky) && \text{(K)}\\
&\equiv KK && (\eta)
\end{aligned}
$$

c)

$$
\begin{aligned}
\lambda xyz.x &\equiv \lambda xy.Kx && \text{(K)}\\
&\equiv \lambda x.K(Kx) && \text{(K)}\\
&\equiv S(\lambda x.K)(\lambda x.Kx) && \text{(S)}\\
&\equiv S(KK)(\lambda x.Kx) && \text{(K)}\\
&\equiv S(KK)K && (\eta)
\end{aligned}
$$

d)

$$\begin{aligned}
\lambda x.fx(fxx) &\equiv S(\lambda x.fx)(\lambda x.fxx) && \text{(S)}\\
&\equiv Sf(\lambda x.fxx) && (\eta)\\
&\equiv Sf(S(\lambda x.fx)(\lambda x.x)) && \text{(S)}\\
&\equiv Sf(Sf(\lambda x.x)) && (\eta)\\
&\equiv Sf(SfI) && \text{(S)}
\end{aligned}$$

e)

$$\begin{aligned}
(\lambda fx.fy)(\lambda x.x) &\equiv \lambda x.(\lambda x.x)y && (\beta)\\
&\equiv \lambda x.y && (\beta)\\
&\equiv Ky && \text{(K)}
\end{aligned}$$