

EXERCISES

CHAPTER 3

SEAN LI ¹

1. Reducted

Definition Some rules for reference.

$$\begin{array}{c} \frac{x : \sigma \in \Gamma \quad \Gamma \text{ is a } \lambda 2 \text{ context}}{\Gamma \vdash x : \sigma} \text{ (T-Var)} \qquad \frac{\Gamma \vdash M : \sigma \rightarrow \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash MN : \tau} \text{ (T-App)} \\[10pt] \frac{\Gamma, x : \sigma \vdash M : \tau}{\Gamma \vdash \lambda x : \sigma. M : \sigma \rightarrow \tau} \text{ (T-Abst)} \qquad \frac{\alpha \in \mathbb{T}_2 \quad \forall \tau \in \text{FV } \alpha, \Gamma \vdash \tau : *}{\alpha : * \in \Gamma} \text{ (T-Form)} \\[10pt] \frac{\Gamma \vdash M : \Pi_{\alpha:*.} A \quad \Gamma \vdash B : *}{\Gamma \vdash MB : A[\alpha := B]} \text{ (T2-App)} \qquad \frac{\Gamma, \alpha : * \vdash M : A}{\Gamma \vdash \lambda \alpha : *. M : \Pi_{\alpha:*.} A} \text{ (T2-Abst)} \end{array}$$

In this document, convention is that all type judgements in a proof tree, unless stated otherwise, is derived from a single and unique $\lambda 2$ context per tree. Multiple conclusions might be drawn on a single line from usage of the same inference rule for compactness. Eg:

ex 1. $\alpha, \beta : *$ **T-Form**

Is shorthand for

ex 1. $\Gamma \vdash \alpha : *$ **T-Form**

ex 2. $\Gamma \vdash \beta : *$ **T-Form**

Problem

(3.1) How many $\lambda 2$ contexts consisting of four and only four declarations

- (1) $\Gamma \vdash \alpha : *$ (2) $\Gamma \vdash \beta : *$
 (3) $\Gamma \vdash f : \alpha \rightarrow \beta$ (4) $\Gamma \vdash x : \alpha$

Solution. The last two declarations depend on the first two. Therefore this is an easy combinatorics problem: $2! \times 2! = 4$ contexts:

- 1 – 2 – 3 – 4 1 – 2 – 4 – 3
 2 – 1 – 3 – 4 2 – 1 – 4 – 3

Problem

(3.2) Give a full derivation in $\lambda 2$ to show the following type term is legal:

$$M \equiv \lambda \alpha, \beta, \gamma : *. \lambda f : \alpha \rightarrow \beta. \lambda g : \beta \rightarrow \gamma. \lambda x : \alpha. g(fx)$$

Solution.

1.	$\alpha : *$	Bound
2.	$\beta : *$	Bound
3.	$\gamma : *$	Bound
4.	$f : \alpha \rightarrow \beta$	Bound
5.	$g : \beta \rightarrow \gamma$	Bound
6.	$x : \alpha$	Bound
7.	$\alpha, \beta, \gamma : *$	T-Form
8.	$\alpha \rightarrow \beta, \beta \rightarrow \gamma : *$	T-Form
9.	$f : \alpha \rightarrow \beta, x : \alpha$	T-Var
10.	$fx : \beta$	8,8 T-App
11.	$g : \beta \rightarrow \gamma$	T-Var
12.	$g(fx) : \gamma$	11,10 T-App
13.	$\lambda x : \alpha. g(fx) : \alpha \rightarrow \gamma$	12 T-Abst
14.	$\lambda g : \beta \rightarrow \gamma. \lambda x : \alpha. g(fx) : (\beta \rightarrow \gamma) \rightarrow \alpha \rightarrow \gamma$	13 T-Abst
15.	$\lambda f : \alpha \rightarrow \beta. \lambda g : \beta \rightarrow \gamma. \lambda x : \alpha. g(fx) : (\alpha \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \alpha \rightarrow \gamma$	14 T-Abst

16.	$\frac{\lambda\gamma : *. \lambda f : \alpha \rightarrow \beta. \lambda g : \beta \rightarrow \gamma. \lambda x : \alpha. g(fx) : \Pi\gamma : *. (\alpha \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \alpha \rightarrow \gamma}{\lambda\beta, \gamma : *. \lambda f : \alpha \rightarrow \beta. \lambda g : \beta \rightarrow \gamma. \lambda x : \alpha. g(fx) : \Pi\beta, \gamma : *. (\alpha \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \alpha \rightarrow \gamma}$	15 T2-Abst
17.		16 T2-Abst
18.		
	$\lambda\alpha, \beta, \gamma : *. \lambda f : \alpha \rightarrow \beta. \lambda g : \beta \rightarrow \gamma. \lambda x : \alpha. g(fx) : \Pi\alpha, \beta, \gamma : *. (\alpha \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \alpha \rightarrow \gamma$	17 T2-Abst

Problem

(3.3 a) Given M in 3.2, and a context Γ such that

$$\Gamma \vdash \text{nat} : *$$

$$\Gamma \vdash \text{bool} : *$$

$$\Gamma \vdash \text{succ} : \text{nat} \rightarrow \text{nat}$$

$$\Gamma \vdash \text{even} : \text{nat} \rightarrow \text{bool}$$

Prove $M \text{ nat nat bool succ even}$ is legal.

Solution. Proof by deriving the term's type.

Proof.

1.	$M : \Pi\alpha, \beta, \gamma : *. (\alpha \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \alpha \rightarrow \gamma$	T-Var
2.	$\text{nat}, \text{bool} : *$	T-Form
3.	$M \text{ nat} : \Pi\beta, \gamma : *. (\text{nat} \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \text{nat} \rightarrow \gamma$	2,1 T2-App
4.	$M \text{ nat nat} : \Pi\gamma : *. (\text{nat} \rightarrow \text{nat}) \rightarrow (\text{nat} \rightarrow \gamma) \rightarrow \text{nat} \rightarrow \gamma$	2,3 T2-App
5.		
	$M \text{ nat nat bool} : (\text{nat} \rightarrow \text{nat}) \rightarrow (\text{nat} \rightarrow \text{bool}) \rightarrow \text{nat} \rightarrow \text{bool}$	2,3 T2-App
6.	$\text{succ} : \text{nat} \rightarrow \text{nat}, \text{even} : \text{nat} \rightarrow \text{bool}$	T-Var
7.	$M \text{ nat nat bool succ} : (\text{nat} \rightarrow \text{bool}) \rightarrow \text{nat} \rightarrow \text{bool}$	6,5 T-App
8.	$M \text{ nat nat bool succ even} : \text{nat} \rightarrow \text{bool}$	6,7 T-App

■

Problem

(3.3 b.i) Prove $\lambda x : \mathbf{nat}. \text{even} (\text{succ } x)$ via 3.3 a.

Solution. The result of beta reduction on the term in 3.3 a is what we are proving.

Proof.

$$\begin{aligned}
 & M \mathbf{nat} \mathbf{nat} \mathbf{bool} \text{succ even} \\
 & \equiv (\lambda \alpha, \beta, \gamma, f, g. \lambda x : \alpha. g(fx)) \mathbf{nat} \mathbf{nat} \mathbf{bool} \text{succ even} \\
 & \xrightarrow[\beta]{\text{}} (\lambda f : \mathbf{nat} \rightarrow \mathbf{nat}. \lambda g : \mathbf{nat} \rightarrow \mathbf{bool}. \lambda x : \mathbf{nat}. g(fx)) \text{succ even} \\
 & \xrightarrow[\beta]{\text{}} (\lambda x : \mathbf{nat}. \text{even} (\text{succ } x))
 \end{aligned}$$

By the subject reduction lemma, $\lambda x : \mathbf{nat}. \text{even} (\text{succ } x) : \mathbf{nat} \rightarrow \mathbf{bool}$, thus is legal. ■

Problem

(3.3 b.ii) Prove $\lambda x : \mathbf{nat}. \text{even} (\text{succ } x)$ via derivation in the context provided in 3.3 a.

Solution.

Proof.

1.	$\mathbf{nat}, \mathbf{bool} : *$	T-Form
2.	$x : \mathbf{nat}$	Bound
3.	$\text{succ} : \mathbf{nat} \rightarrow \mathbf{nat}$	T-Var
4.	$x : \mathbf{nat}$	T-Var
5.	$\text{succ } x : \mathbf{nat}$	3,4 T-App
6.	$\text{even} : \mathbf{nat} \rightarrow \mathbf{bool}$	T-Var
7.	$\text{even} (\text{succ } x) : \mathbf{bool}$	6,5 T-App
8.	$\lambda x : \mathbf{nat}. \text{even} (\text{succ } x) : \mathbf{nat} \rightarrow \mathbf{bool}$	7 T-Abst

■

Problem

(3.4) Give a shorthand (omit T-Var and T-Form) derivation in $\lambda 2$ to show the following term is valid in $\Gamma \equiv \text{nat} : *, \text{bool} : *$

$$(\lambda \alpha, \beta : *. \lambda f : \alpha \rightarrow \alpha. \lambda g : \alpha \rightarrow \beta. \lambda x : \alpha. g(f(fx))) \text{ nat } \text{bool}$$

Solution.

Proof.

1.	$\alpha, \beta : *$	Bound
2.	$f : \alpha \rightarrow \alpha$	Bound
3.	$g : \alpha \rightarrow \beta$	Bound
4.	$x : \alpha$	Bound
5.	$fx : \alpha$	*, T-App
6.	$f(fx) : \alpha$	*,5 T-App
7.	$g(f(fx)) : \beta$	*,6 T-App
8.	$\lambda x : \alpha. g(f(fx)) : \alpha \rightarrow \beta$	7 T-Abst
9.	$\lambda g : \alpha \rightarrow \beta. \lambda x : \alpha. g(f(fx)) : (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \beta$	8 T-Abst
	$\lambda f : \alpha \rightarrow \alpha. \lambda g : \alpha \rightarrow \beta. \lambda x : \alpha. g(f(fx))$	
10.	$: (\alpha \rightarrow \alpha) \rightarrow (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \beta$	9 T-Abst
11.		
	$\lambda \alpha, \beta : *. \lambda f : \alpha \rightarrow \alpha. \lambda g : \alpha \rightarrow \beta. \lambda x : \alpha. g(f(fx))$	
	$: \Pi \alpha, \beta : *. (\alpha \rightarrow \alpha) \rightarrow (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \beta$	10 T2-Abst
12.		
	$(\lambda \alpha, \beta : *. \lambda f : \alpha \rightarrow \alpha. \lambda g : \alpha \rightarrow \beta. \lambda x : \alpha. g(f(fx))) \text{ nat}$	
	$: \Pi \beta : *. (\text{nat} \rightarrow \text{nat}) \rightarrow (\text{nat} \rightarrow \beta) \rightarrow \text{nat} \rightarrow \beta$	*,11 T2-App
13.		
	$(\lambda \alpha, \beta : *. \lambda f : \alpha \rightarrow \alpha. \lambda g : \alpha \rightarrow \beta. \lambda x : \alpha. g(f(fx))) \text{ nat } \text{bool}$	
	$: \Pi \beta : *. (\text{nat} \rightarrow \text{nat}) \rightarrow (\text{nat} \rightarrow \text{bool}) \rightarrow \text{nat} \rightarrow \text{bool}$	*,12 T2-App

■

Problem

(3.5 a) Let $\perp \equiv \Pi \alpha : *. \alpha$. Prove \perp is legal.

Solution. Here a notion called kind checking is introduced. This has not yet been discussed in this book (?)

Proof.

1. $\alpha : *$ **Bound**
2. $\frac{}{\alpha : *}$ **T-Form**
3. $\Pi \alpha : *. \alpha : * \rightarrow *$ **T-Form**

■

Problem

(3.5 b) Consider the context $\Gamma \equiv \beta : *, x : \perp$. Find an inhabitant of type β under Γ .

Solution. $x\beta$ is. Because x is of second-order type, it must be parametric to a type, thus x is of form $\lambda \alpha : *. M$ where $\Gamma, \alpha : * \vdash M : \alpha$.

Proof.

1. $x : \Pi \alpha : *. \alpha$ **T-Var**
2. $\beta : *$ **T-Form**
3. $x\beta : \beta$ **1,2 T2-App**

■

Problem

(3.5 c) Give three inhabitants of $\beta \rightarrow \beta$ in β -nf under Γ in 3.5 b.

Solution.

1. $\lambda y : \beta. y$.

Proof.

1. $y : \beta$ **Bound**
2. $\frac{}{y : \beta}$ **T-Var**
3. $\lambda y : \beta. y : \beta \rightarrow \beta$ **2 T-Abst**

■

2. $\lambda y : \beta. x\beta$.

Proof.

- | | | |
|----|---|-------------------|
| 1. | $y : \beta$ | Bound |
| 2. | $x : \Pi\alpha : * . \alpha$ | T-Var |
| 3. | $\beta : *$ | T-Form |
| 4. | $x\beta : \beta$ | 2,3 T2-App |
| 5. | $\lambda y : \beta. x\beta : \beta \rightarrow \beta$ | 4 T-Abst |

■

3. $\lambda y : \beta. x(\beta \rightarrow \beta)y.$

Proof.

- | | | |
|----|--|-------------------|
| 1. | $y : \beta$ | Bound |
| 2. | $x : \Pi\alpha : * . \alpha$ | T-Var |
| 3. | $\beta \rightarrow \beta : *$ | T-Form |
| 4. | $x(\beta \rightarrow \beta) : \beta \rightarrow \beta$ | 2,3 T2-App |
| 5. | $y : \beta$ | T-Var |
| 6. | $x(\beta \rightarrow \beta)y : \beta$ | 4,5 T-App |
| 7. | $\lambda y : \beta. x(\beta \rightarrow \beta)y : \beta \rightarrow \beta$ | 5 T-Abst |

■

Problem

(3.5 d) Prove that the following terms inhabit the same type in Γ :

$$\lambda f : \beta \rightarrow \beta \rightarrow \beta. f(x\beta)(x\beta)$$

$$x((\beta \rightarrow \beta \rightarrow \beta) \rightarrow \beta)$$

Solution. We simply derive the types.

First Term.

- | | | |
|----|---|-------------------|
| 1. | $f : \beta \rightarrow \beta \rightarrow \beta$ | Bound |
| 2. | $f : \beta \rightarrow \beta \rightarrow \beta$ | T-Var |
| 3. | $x : \Pi\alpha : * . \alpha$ | T-Var |
| 4. | $\beta : *$ | T-Form |
| 5. | $x\beta : \beta$ | 3,4 T2-App |
| 6. | $f(x\beta) : \beta \rightarrow \beta$ | 2,5 T-App |

7. $\frac{}{f(x\beta)(x\beta) : \beta}$ **6,5 T-App**
8. $\lambda f : \beta \rightarrow \beta \rightarrow \beta. f(x\beta)(x\beta) : (\beta \rightarrow \beta \rightarrow \beta) \rightarrow \beta$ **6 T-Abst**

■

Second Term.

1. $(\beta \rightarrow \beta \rightarrow \beta) \rightarrow \beta : *$ **T-Form**
2. $x : \Pi \alpha : *. \alpha$ **T-Var**
3. $x((\beta \rightarrow \beta \rightarrow \beta) \rightarrow \beta) : (\beta \rightarrow \beta \rightarrow \beta) \rightarrow \beta$ **2,1 T2-App**

■

The two terms were shown to both inhabit $(\beta \rightarrow \beta \rightarrow \beta) \rightarrow \beta$.

Problem

(3.6 a) Find inhabitant of type

$$\Pi \alpha, \beta : *. (\text{nat} \rightarrow \alpha) \rightarrow (\alpha \rightarrow \text{nat} \rightarrow \beta) \rightarrow \text{nat} \rightarrow \beta$$

In context $\Gamma \equiv \text{nat} : *$.

Solution.

$$\lambda \alpha, \beta : *. \lambda x : \text{nat} \rightarrow \alpha. \lambda y : (\alpha \rightarrow \text{nat} \rightarrow \beta). \lambda z : \text{nat}. y(xz)z$$

Proof.

1. $\alpha, \beta : *$ **Bound**
2. $\text{nat} \rightarrow \alpha : *$ **T-Form**
3. $x : \text{nat} \rightarrow \alpha$ **Bound**
4. $\alpha \rightarrow \text{nat} \rightarrow \beta : *$ **T-Form**
5. $y : \alpha \rightarrow \text{nat} \rightarrow \beta$ **Bound**
6. $\text{nat} : *$ **Bound**
7. $z : \text{nat}$ **Bound**
8. $y : \alpha \rightarrow \text{nat} \rightarrow \beta$ **T-Var**
9. $x : \text{nat} \rightarrow \alpha$ **T-Var**
10. $z : \text{nat}$ **T-Var**
11. $xz : \alpha$ **9,10 T-App**
12. $y(xz) : \text{nat} \rightarrow \beta$ **8,11 T-App**
13. $y(xz)z : \beta$ **12,10 T-App**

14.	$\lambda z : \mathbf{nat}. y(xz)z : \mathbf{nat} \rightarrow \beta$	13 T-Abst
15.	$\lambda y : \alpha \rightarrow \mathbf{nat} \rightarrow \beta. \lambda z : \mathbf{nat}. y(xz)z$ $: (\alpha \rightarrow \mathbf{nat} \rightarrow \beta) \rightarrow \mathbf{nat} \rightarrow \beta$	14 T-Abst
16.	$\lambda x : \mathbf{nat} \rightarrow \alpha. y : \alpha \rightarrow \mathbf{nat} \rightarrow \beta. \lambda z : \mathbf{nat}. y(xz)z$ $: (\mathbf{nat} \rightarrow \alpha) \rightarrow (\alpha \rightarrow \mathbf{nat} \rightarrow \beta) \rightarrow \mathbf{nat} \rightarrow \beta$	15 T2-Abst
17.		
	$\lambda \alpha, \beta : *. x : \mathbf{nat} \rightarrow \alpha. y : \alpha \rightarrow \mathbf{nat} \rightarrow \beta. \lambda z : \mathbf{nat}. y(xz)z$ $: \Pi \alpha, \beta : *. (\mathbf{nat} \rightarrow \alpha) \rightarrow (\alpha \rightarrow \mathbf{nat} \rightarrow \beta) \rightarrow \mathbf{nat} \rightarrow \beta$	16 T2-Abst

■

Problem

(3.6 b) Find inhabitant of type

$$\Pi \delta : *. ((\alpha \rightarrow \gamma) \rightarrow \delta) \rightarrow (\alpha \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \delta$$

In context $\Gamma \equiv \alpha : *, \beta : *, \gamma : *$

Solution.

$$\lambda \delta : *. \lambda x : (\alpha \rightarrow \gamma) \rightarrow \delta. \lambda y : \alpha \rightarrow \beta. \lambda z : \beta \rightarrow \gamma. x(\lambda u : \alpha. z(yu))$$

A derivation in shorthand will be given (omitting T-Form / T-Var)

Proof.

1.	$\delta : *$	Bound
2.	$x : (\alpha \rightarrow \gamma) \rightarrow \delta$	Bound
3.	$y : \alpha \rightarrow \beta$	Bound
4.	$z : \beta \rightarrow \gamma$	Bound
5.	$u : \alpha$	Bound
6.	$yu : \beta$	*, T-App
7.	$z(yu) : \gamma$	*,6 T-App
8.	$\lambda u : \alpha. z(yu) : \alpha \rightarrow \gamma$	7 T-Abst
9.	$x(\lambda u : \alpha. z(yu)) : \delta$	8 T-Abst
10.	$\lambda z : \beta \rightarrow \gamma. x(\lambda u : \alpha. z(yu)) : (\beta \rightarrow \gamma) \rightarrow \delta$	9 T-Abst
	$\lambda y : \alpha \rightarrow \beta. \lambda z : \beta \rightarrow \gamma. x(\lambda u : \alpha. z(yu))$	
11.	$: (\alpha \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \delta$	10 T-Abst

12. $\lambda x : (\alpha \rightarrow \gamma) \rightarrow \delta. \lambda y : \alpha \rightarrow \beta. \lambda z : \beta \rightarrow \gamma. x(\lambda u : \alpha. z(yu))$ **11 T-Abst**
 13. $: ((\alpha \rightarrow \gamma) \rightarrow \beta) \rightarrow (\alpha \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \delta$
- $\lambda \delta : *. \lambda x : (\alpha \rightarrow \gamma) \rightarrow \delta. \lambda y : \alpha \rightarrow \beta. \lambda z : \beta \rightarrow \gamma. x(\lambda u : \alpha. z(yu))$
 $: \Pi \delta : *. ((\alpha \rightarrow \gamma) \rightarrow \delta) \rightarrow (\alpha \rightarrow \beta) \rightarrow (\beta \rightarrow \gamma) \rightarrow \delta$ **12 T2-Abst**

■

Problem

(3.6 c) Find inhabitant of type

$$\Pi \alpha, \beta, \gamma : *. (\alpha \rightarrow (\beta \rightarrow \alpha) \rightarrow \gamma) \rightarrow \alpha \rightarrow \gamma$$

In the empty context

Solution.

$$\lambda \alpha, \beta, \gamma : *. \lambda f : (\alpha \rightarrow (\beta \rightarrow \alpha) \rightarrow \gamma). \lambda x : \alpha. fx(\lambda u : \beta. x)$$

Proof.

- | | | |
|-----|--|------------------|
| 1. | α, β, γ | Bound |
| 2. | $f : \alpha \rightarrow (\beta \rightarrow \alpha) \rightarrow \gamma$ | Bound |
| 3. | $x : \alpha$ | Bound |
| 4. | $fx : (\beta \rightarrow \alpha) \rightarrow \gamma$ | *, T-App |
| 5. | $u : \beta$ | Bound |
| 6. | $x : \alpha$ | T-Var |
| 7. | $\lambda u : \beta. x : \beta \rightarrow \alpha$ | 6 T-Abst |
| 8. | $fx(\lambda u : \beta. x) : \gamma$ | 4,7 T-App |
| 9. | $\lambda x : \alpha. fx(\lambda u : \beta. x) : \alpha \rightarrow \gamma$ | 8 T-Abst |
| 10. | $\lambda f : \alpha \rightarrow (\beta \rightarrow \alpha) \rightarrow \gamma. \lambda x : \alpha. fx(\lambda u : \beta. x)$
$: (\alpha \rightarrow (\beta \rightarrow \alpha) \rightarrow \gamma) \rightarrow \alpha \rightarrow \gamma$ | 9 T-Abst |
| 11. | | |
- $\lambda \alpha, \beta, \gamma : *. \lambda f : \alpha \rightarrow (\beta \rightarrow \alpha) \rightarrow \gamma. \lambda x : \alpha. fx(\lambda u : \beta. x)$
 $: \Pi \alpha, \beta, \gamma : *. (\alpha \rightarrow (\beta \rightarrow \alpha) \rightarrow \gamma) \rightarrow \alpha \rightarrow \gamma$ **10 T2-Abst**

■

Problem

(3.7) Let $\perp \equiv \Pi\alpha : * . \alpha$ and context $\Gamma \equiv \alpha : *, \beta : *, x : \alpha \rightarrow \perp, f : (\alpha \rightarrow \alpha) \rightarrow \alpha$. Give a derivation that succesively calculate an inhabitant of α and β , both in context Γ .

Solution. Have $M : \alpha := f(\lambda n : \alpha. n)$. Then $\Gamma \vdash xM\beta : \beta$.

Typing M .

1. $f : (\alpha \rightarrow \alpha) \rightarrow \alpha$ **T-Var**
2. $n : \alpha$ **Bound**
3. $\frac{}{n : \alpha}$ **T-Var**
4. $\lambda n : \alpha. n : \alpha \rightarrow \alpha$ **3 T-Abst**
5. $f(\lambda n : \alpha. n) : \alpha$ **1,4 T-App**

■

Typing $xM\beta$.

1. $M : \alpha$ **T-Var**
2. $x : \alpha \rightarrow \Pi\alpha : * . \alpha$ **T-Var**
3. $Mx : \Pi\alpha : * . \alpha$ **2,1 T-App**
4. $Mx\beta : \beta$ **3,* T2-App**

■

Problem

(3.8) Recall $K \equiv \lambda xy. x \in \Lambda$ from untyped lambda calculus. Consider the following types

$$T_1 \equiv \Pi\alpha, \beta : * . \alpha \rightarrow \beta \rightarrow \alpha \quad T_2 \equiv \Pi\alpha : * . \alpha \rightarrow (\Pi\beta : * . \beta \rightarrow \alpha)$$

Find inhabitants of both type $t_1 : T_1$ and $t_2 : T_2$ under the empty context, which may be considered the closed $\lambda 2$ form of $K \in \Lambda_{\mathbb{T}2}$.

Solution.

$$\lambda\alpha, \beta : * . \lambda x : \alpha. \lambda y : \beta. x$$

$$\lambda\alpha : * . \lambda x : \alpha. \lambda\beta : * . \lambda y : \beta. x$$

First Form.

1.	$\alpha, \beta : *$	Bound
2.	$x : \alpha$	Bound
3.	$y : \beta$	Bound
4.	$x : \alpha$	T-Var
5.	$\lambda y : \beta. x : \beta \rightarrow \alpha$	4 T-Abst
6.	$\lambda x : \alpha. \lambda y : \beta. x : \beta \rightarrow \alpha$	5 T-Abst
7.	$\lambda \alpha, \beta : *. \lambda x : \alpha. \lambda y : \beta. x : \beta \rightarrow \alpha \rightarrow \beta \rightarrow \alpha$	5 T2-Abst

■

Second Form.

1.	$\alpha : *$	Bound
2.	$x : \alpha$	Bound
3.	$\beta : *$	Bound
4.	$y : \beta$	Bound
5.	$x : \alpha$	T-Var
6.	$\lambda y : \beta. x : \beta \rightarrow \alpha$	5 T-Abst
7.	$\lambda \beta : *. \lambda y : \beta. x : \beta \rightarrow \alpha$	6 T2-Abst
8.	$\lambda x : \alpha. \lambda \beta : *. \lambda y : \beta. x : \beta \rightarrow \alpha \rightarrow (\Pi \beta : *. \beta \rightarrow \alpha)$	7 T-Abst
9.	$\lambda \alpha : *. \lambda x : \alpha. \lambda \beta : *. \lambda y : \beta. x : \beta \rightarrow \alpha \rightarrow (\Pi \alpha : *. \alpha \rightarrow (\Pi \beta : *. \beta \rightarrow \alpha))$	8 T2-Abst

■

Problem

(3.9) Pretype the combinator

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

In closed form (typable in an empty context) in Λ_{T2} .

Solution.

$$S \equiv \lambda \alpha, \beta, \gamma : *. \lambda x : \alpha \rightarrow \beta \rightarrow \gamma. \lambda y : \alpha \rightarrow \beta. \lambda z : \alpha. x z (y z)$$

Proof.

1.	$\alpha, \beta, \gamma : *$	Bound
2.	$x : \alpha \rightarrow \beta \rightarrow \gamma$	Bound
3.	$y : \alpha \rightarrow \beta$	Bound

4.			$z : \alpha$	Bound
5.			$xz : \beta \rightarrow \gamma$	*,* T-App
6.			$yx : \beta$	*,* T-App
7.			$xz(yx) : \gamma$	5,6 T-App
8.			$\lambda z : \alpha.xz(yx) : \alpha \rightarrow \gamma$	7 T-Abst
9.			$\lambda y : \alpha \rightarrow \beta.\lambda z : \alpha.xz(yx) : (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \gamma$	8 T-Abst
10.			$\lambda x : \alpha \rightarrow \beta \rightarrow \gamma.\lambda y : \alpha \rightarrow \beta.\lambda z : \alpha.xz(yx)$ $: (\alpha \rightarrow \beta \rightarrow \gamma) \rightarrow (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \gamma$	9 T-Abst
11.				
			$\lambda \alpha, \beta, \gamma : *. \lambda x : \alpha \rightarrow \beta \rightarrow \gamma.\lambda y : \alpha \rightarrow \beta.\lambda z : \alpha.xz(yx)$ $: \Pi \alpha, \beta, \gamma : *. (\alpha \rightarrow \beta \rightarrow \gamma) \rightarrow (\alpha \rightarrow \beta) \rightarrow \alpha \rightarrow \gamma$	10 T2-Abst

■

Problem

(3.10 a) Consider the term

$$M \equiv \lambda x : \Pi \alpha : *. \alpha \rightarrow \alpha.x(\sigma \rightarrow \sigma)(x\sigma)$$

Prove that M is legal.

Solution. For a term to be legal there must exist a context so that the term could be typed. Here, a witness context is $\Gamma \equiv \sigma : *$.

Proof.

1.	$x : \Pi \alpha : *. \alpha \rightarrow \alpha$	Bound
2.	$x(\sigma \rightarrow \sigma) : (\sigma \rightarrow \sigma) \rightarrow (\sigma \rightarrow \sigma)$	*,* T2-App
3.	$x\sigma : \sigma \rightarrow \sigma$	*,* T2-App
4.	$x(\sigma \rightarrow \sigma)(x\sigma) : \sigma \rightarrow \sigma$	2,3 T-App
5.	$\lambda x : \Pi \alpha : *. \alpha \rightarrow \alpha.x(\sigma \rightarrow \sigma)(x\sigma) : (\Pi \alpha : *. \alpha \rightarrow \alpha) \rightarrow \sigma \rightarrow \sigma$	4 T-Abst

■

Problem

(3.10 b) Find a term N such that MN is legal and may be considered to be a way to add type information to $(\lambda x.xx)(\lambda y.y)$

Solution.

$$M\sigma N \equiv (\lambda x : \Pi\alpha : * . \alpha \rightarrow \alpha . x(\sigma \rightarrow \sigma)(x\sigma))\sigma(\lambda y : \sigma . y)$$

Is the same as $(\lambda x . xx)(\lambda y . y)$ modulo type annotations.

Proof.

1. $M : (\Pi\alpha : * . \alpha \rightarrow \alpha) \rightarrow \sigma \rightarrow \sigma$ **T-Var**
2. $M\sigma : (\sigma \rightarrow \sigma) \rightarrow \sigma \rightarrow \sigma$ **1,* T2-App**
3. $y : \sigma$ **Bound**
4. $\boxed{y : \sigma}$ **T-Var**
5. have $N := \lambda y : \sigma . y : \sigma \rightarrow \sigma$ **4 T-Abst**
6. $M\sigma N : \sigma \rightarrow \sigma$ **2,5 T-Abst**

■

Problem

(3.11) Recall $\perp \equiv \Pi\alpha : * . \alpha$ from 3.5. Type and prove the following term legal:

$$\lambda x : \perp . x(\perp \rightarrow \perp \rightarrow \perp)(x(\perp \rightarrow \perp)x)(x(\perp \rightarrow \perp \rightarrow \perp)xx)$$

Solution.

Proof. The type $\perp \rightarrow \perp$ is closed and well formed. Therefore, the term is legal.

1. $\perp : * \equiv \Pi\alpha : * . \alpha$ **T-Form**
2. $x : \perp$ **Bound**
3. $\boxed{x(\perp \rightarrow \perp \rightarrow \perp) : \perp \rightarrow \perp \rightarrow \perp}$ ***,* T2-App**
4. $\boxed{x(\perp \rightarrow \perp) : \perp \rightarrow \perp}$ ***,* T2-App**
5. $\boxed{x(\perp \rightarrow \perp)x : \perp}$ **4,* T-App**
6. $\boxed{x(\perp \rightarrow \perp \rightarrow \perp)(x(\perp \rightarrow \perp)x) : \perp \rightarrow \perp}$ **3,5 T-App**
7. $\boxed{x(\perp \rightarrow \perp \rightarrow \perp)x : \perp \rightarrow \perp}$ **3,* T-App**
8. $\boxed{x(\perp \rightarrow \perp \rightarrow \perp)xx : \perp}$ **7,* T-App**
9. $\boxed{x(\perp \rightarrow \perp \rightarrow \perp)(x(\perp \rightarrow \perp)x)(x(\perp \rightarrow \perp \rightarrow \perp)xx) : \perp}$ **6,8 T-App**
10. $\lambda x : \perp . x(\perp \rightarrow \perp \rightarrow \perp)(x(\perp \rightarrow \perp)x)(x(\perp \rightarrow \perp \rightarrow \perp)xx) : \perp$ **9 T-Abst**

■

Problem

(3.12) Given the Polymorphic Church Numerals:

$$\mathbf{nat} \in \mathbb{T}_2 := \Pi \alpha : *. (\alpha \rightarrow \alpha) \rightarrow \alpha \rightarrow \alpha$$

$$\bar{0} \equiv \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. x : \mathbf{nat}$$

$$\bar{1} \equiv \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. f x : \mathbf{nat}$$

$$\bar{2} \equiv \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. f(f x) : \mathbf{nat}$$

$$\mathbf{succ} \equiv \lambda n : \mathbf{nat}. \lambda \beta : *. \lambda f : \beta \rightarrow \beta. \lambda x : \beta. f(n \beta f x)$$

Prove that

$$\mathbf{succ} \bar{0} \underset{\beta}{=} \bar{1}$$

$$\mathbf{succ} \bar{1} \underset{\beta}{=} \bar{2}$$

Solution.

$$\begin{aligned} & \mathbf{succ} \bar{0} \\ & \equiv (\lambda n : \mathbf{nat}. \lambda \beta : *. \lambda f : \beta \rightarrow \beta. \lambda x : \beta. f(n \beta f x)) (\lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. x) \\ & \xrightarrow{\beta} (\lambda \beta : *. \lambda f : \beta \rightarrow \beta. \lambda x : \beta. f((\lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. x) \beta f x)) \\ & \xrightarrow{\beta_{\mathbb{T}_2}} (\lambda \beta : *. \lambda f : \beta \rightarrow \beta. \lambda x : \beta. f((\lambda f : \beta \rightarrow \beta. \lambda x : \beta. x) f x)) \\ & \xrightarrow[\beta]{\beta \rightarrow \alpha} (\lambda \beta : *. \lambda f : \beta \rightarrow \beta. \lambda x : \beta. f x) \underset{\alpha_{\mathbb{T}_2}}{\equiv} \bar{1} \\ & \mathbf{succ} \bar{1} \\ & \equiv (\lambda n : \mathbf{nat}. \lambda \beta : *. \lambda f : \beta \rightarrow \beta. \lambda x : \beta. f(n \beta f x)) (\lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. f x) \\ & \xrightarrow{\beta} (\lambda \beta : *. \lambda f : \beta \rightarrow \beta. \lambda x : \beta. f((\lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. f x) \beta f x)) \\ & \xrightarrow{\beta_{\mathbb{T}_2}} (\lambda \beta : *. \lambda f : \beta \rightarrow \beta. \lambda x : \beta. f((\lambda f : \beta \rightarrow \beta. \lambda x : \beta. f x) f x)) \\ & \xrightarrow[\beta]{\beta \rightarrow \alpha} (\lambda \beta : *. \lambda f : \beta \rightarrow \beta. \lambda x : \beta. f(f x)) \underset{\alpha_{\mathbb{T}_2}}{\equiv} \bar{2} \end{aligned}$$

Problem

(3.13 a) We define addition in Polymorphic Church Numerals as

$$\text{add} \equiv \lambda m, n : \mathbf{nat}. \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \mathbf{nat}. m \alpha f (n \alpha f x)$$

Show that

$$\text{add } \bar{1} \quad \bar{1} \underset{\beta}{=} \bar{2}$$

Solution.

$$\begin{aligned} & \text{add } \bar{1} \quad \bar{1} \\ \equiv & (\lambda m, n : \mathbf{nat}. \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. m \alpha f (n \alpha f x)) \bar{1} \quad \bar{1} \\ \xrightarrow[\beta]{} & \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. \bar{1} \alpha f (\bar{1} \alpha f x) \\ \equiv & \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. \bar{1} \alpha f ((\lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. f x) \alpha f x) \\ \xrightarrow[\beta]{} & \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. \bar{1} \alpha f (f x) \\ \equiv & \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. (\lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. f x) \alpha f (f x) \\ \xrightarrow[\beta]{} & \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. f (f x) \underset{\alpha}{=} \bar{2} \end{aligned}$$

Problem

(3.13 b) Find a term `mul` simulates multiplication on `nat`.

Solution.

$$\text{mul} \equiv \lambda m, n : \mathbf{nat}. \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. m \alpha (n \alpha f) x$$

Proof. We derive the type first to prove a legal term.

1.	$m, n : \mathbf{nat}$	Bound
2.	$\alpha : *$	Bound
3.	$f : \alpha \rightarrow \alpha$	Bound
4.	$x : \alpha$	Bound
5.	$m \alpha : (\alpha \rightarrow \alpha) \rightarrow \alpha \rightarrow \alpha$	*, T2-App
6.	$n \alpha : (\alpha \rightarrow \alpha) \rightarrow \alpha \rightarrow \alpha$	*, T2-App
7.	$n \alpha f : \alpha \rightarrow \alpha$	6, T-App
8.	$m \alpha (n \alpha f) : \alpha \rightarrow \alpha$	5,7 T-App

9.	$\frac{}{m\alpha(n\alpha f)x : \alpha}$	8,* T-App
10.	$\frac{}{\lambda x : \alpha. m\alpha(n\alpha f)x : \alpha \rightarrow \alpha}$	9 T-Abst
11.	$\frac{}{\lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. m\alpha(n\alpha f)x : (\alpha \rightarrow \alpha) \rightarrow \alpha \rightarrow \alpha}$	10 T-Abst
12.	$\frac{}{\lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. m\alpha(n\alpha f)x : \mathbf{nat}}$	11 T2-Abst
13.		

$\lambda m, n : \mathbf{nat}. \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. m\alpha(n\alpha f)x$
 $: \mathbf{nat} \rightarrow \mathbf{nat} \rightarrow \mathbf{nat}$ **12 T-Abst**

This proves that the term does indeed produce a natural number from two. Next let's prove that

$$\forall \bar{n}, \bar{m} : \mathbf{nat} \quad \text{mul } \bar{n} \ \bar{m} = \overline{n \times m}$$

It could be proven by induction that

$$\forall \bar{a} : \mathbf{nat} \quad \bar{a} \equiv \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. f^a x$$

$$\begin{aligned} \text{mul } \bar{n} \ \bar{m} &\equiv (\lambda m, n : \mathbf{nat}. \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. m\alpha(n\alpha f)x)(\bar{n})(\bar{m}) \\ &\xrightarrow[\beta]{} \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. \bar{n}\alpha(\bar{m}\alpha f)x \\ &\xrightarrow[\beta]{} \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. \bar{n}\alpha(\lambda u : \alpha. f^m u)x \\ &\xrightarrow[\beta]{} \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. (\lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. f^n x)(\lambda u : \alpha. f^m u)x \\ &\xrightarrow[\beta]{} \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. (\lambda u : \alpha. f^m u)^n x \end{aligned}$$

By induction this can be further beta-reduced to

$$\xrightarrow[\beta]{} \lambda \alpha : *. \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. f^{mn} x \equiv \overline{mn}$$

■

Problem

(3.14) We present the Church-Encoded Boolean:

$$\mathbf{bool} \in \mathbb{T}_2 := \Pi \alpha : *. \alpha \rightarrow \alpha \rightarrow \alpha$$

$$\mathbf{true} \equiv \lambda \alpha : *. \lambda x, y : \alpha. x$$

$$\mathbf{false} \equiv \lambda \alpha : *. \lambda x, y : \alpha. y$$

Construct a $\lambda 2$ term \mathbf{neg} such that $\mathbf{neg} \ \mathbf{true} \equiv_{\beta} \mathbf{false}$ and $\mathbf{neg} \ \mathbf{false} \equiv_{\beta} \mathbf{true}$.

Solution.

$$\mathbf{neg} \equiv \lambda b : \mathbf{bool}. \lambda \alpha : *. b\alpha(\mathbf{false} \ \alpha)(\mathbf{true} \ \alpha)$$

Neg True.

$$\begin{aligned}
 \text{neg true} &\equiv (\lambda b : \text{bool}. \lambda \alpha : * . b \alpha (\text{false } \alpha) (\text{true } \alpha)) \text{ true} \\
 &\xrightarrow[\beta]{\rightarrow} \lambda \alpha : * . (\lambda x, y : \alpha . x) (\text{false } \alpha) (\text{true } \alpha) \\
 &\xrightarrow[\beta]{\rightarrow} \lambda \alpha : * . \text{false } \alpha \xrightarrow[\eta]{\rightarrow} \text{false}
 \end{aligned}$$

■

Neg False.

$$\begin{aligned}
 \text{neg false} &\equiv (\lambda b : \text{bool}. \lambda \alpha : * . b \alpha (\text{false } \alpha) (\text{true } \alpha)) \text{ false} \\
 &\xrightarrow[\beta]{\rightarrow} \lambda \alpha : * . (\lambda x, y : \alpha . y) (\text{false } \alpha) (\text{true } \alpha) \\
 &\xrightarrow[\beta]{\rightarrow} \lambda \alpha : * . \text{true } \alpha \xrightarrow[\eta]{\rightarrow} \text{true}
 \end{aligned}$$

■

Problem

(3.15) Define

$$M \equiv \lambda u, v : \text{bool}. \lambda \beta : * . \lambda x, y : \beta . u \beta (v \beta x y) (v \beta y y)$$

And reduce $M \text{ true true}$, $M \text{ true false}$, $M \text{ false true}$, $M \text{ false false}$, and decide which logical operator is represented by M .

Solution.

$$\begin{aligned}
 M \text{ true true} &\equiv (\lambda u, v : \text{bool}. \lambda \beta : * . \lambda x, y : \beta . u \beta (v \beta x y) (v \beta y y)) \text{ true true} \\
 &\xrightarrow[\beta]{\rightarrow} \lambda \beta : * . \lambda x, y : \beta . \text{true } \beta (\text{true } \beta x y) (\text{true } \beta y y) \\
 &\xrightarrow[\beta]{\rightarrow} \lambda \beta : * . \lambda x, y : \beta . (\text{true } \beta x y) \\
 &\xrightarrow[\beta]{\rightarrow} \lambda \beta : * . \lambda x, y : \beta . x \equiv_{\alpha} \text{true}
 \end{aligned}$$

$$\begin{aligned}
 M \text{ true false} &\equiv (\lambda u, v : \text{bool}. \lambda \beta : * . \lambda x, y : \beta . u \beta (v \beta x y) (v \beta y y)) \text{ true false} \\
 &\xrightarrow[\beta]{\rightarrow} \lambda \beta : * . \lambda x, y : \beta . \text{true } \beta (\text{false } \beta x y) (\text{false } \beta y y) \\
 &\xrightarrow[\beta]{\rightarrow} \lambda \beta : * . \lambda x, y : \beta . (\text{false } \beta x y) \\
 &\xrightarrow[\beta]{\rightarrow} \lambda \beta : * . \lambda x, y : \beta . y \equiv_{\alpha} \text{false}
 \end{aligned}$$

$$\begin{aligned}
M \text{ false true} &\equiv (\lambda u, v : \text{bool}. \lambda \beta : *. \lambda x, y : \beta. u \beta (v \beta x y) (v \beta y y)) \text{ false true} \\
&\xrightarrow[\beta]{\rightarrow} \lambda \beta : *. \lambda x, y : \beta. \text{false } \beta (\text{true } \beta x y) (\text{true } \beta y y) \\
&\xrightarrow[\beta]{\rightarrow} \lambda \beta : *. \lambda x, y : \beta. (\text{true } \beta y y) \\
&\xrightarrow[\beta]{\rightarrow} \lambda \beta : *. \lambda x, y : \beta. y \equiv_{\alpha} \text{false}
\end{aligned}$$

$$\begin{aligned}
M \text{ false false} &\equiv (\lambda u, v : \text{bool}. \lambda \beta : *. \lambda x, y : \beta. u \beta (v \beta x y) (v \beta y y)) \text{ false false} \\
&\xrightarrow[\beta]{\rightarrow} \lambda \beta : *. \lambda x, y : \beta. \text{false } \beta (\text{false } \beta x y) (\text{false } \beta y y) \\
&\xrightarrow[\beta]{\rightarrow} \lambda \beta : *. \lambda x, y : \beta. (\text{false } \beta y y) \\
&\xrightarrow[\beta]{\rightarrow} \lambda \beta : *. \lambda x, y : \beta. y \equiv_{\alpha} \text{false}
\end{aligned}$$

Therefore M is equivalent to logical AND.

Problem

(3.16) Find $\lambda 2$ term representing the logical OR, XOR, IMP.

Solution.

$$\begin{aligned}
\text{OR} &\equiv \lambda u, v : \text{bool}. \lambda \beta : *. \lambda x, y : \beta. u \beta x (v \beta x y) \\
\text{XOR} &\equiv \lambda u, v : \text{bool}. \lambda \beta : *. \lambda x, y : \beta. u \beta (v \beta y x) (v \beta x y) \\
\text{IMP} &\equiv \lambda u, v : \text{bool}. \lambda \beta : *. \lambda x, y : \beta. u \beta (v \beta x y) x
\end{aligned}$$

All of them could be checked by finite enumeration over $\text{bool} \times \text{bool}$.

Problem

(3.17) Find $\text{isZero} : \text{nat} \rightarrow \text{bool}$ such that $\forall n : \text{nat}, \text{isZero } n \equiv_{\beta} \text{false}$ except when $n \equiv \bar{0}$.

Solution.

$$\text{isZero} \equiv \lambda n : \text{nat}. n \text{ bool } (\lambda u : \text{bool}. \text{false}) \text{ true}$$

Proof.

$$\begin{aligned}
\text{isZero } \bar{0} &\equiv (\lambda n : \mathbf{nat}. n \text{ bool } (\lambda u : \mathbf{bool}. \text{false}) \text{ true}) \bar{0} \\
&\xrightarrow[\beta]{} (\lambda \alpha : * . \lambda f : \alpha \rightarrow \alpha. \lambda x : \alpha. x) \text{ bool } (\lambda u : \mathbf{bool}. \text{false}) \text{ true} \\
&\xrightarrow[\beta]{} (\lambda f : \mathbf{bool} \rightarrow \mathbf{bool}. \lambda x : \mathbf{bool}. x) (\lambda u : \mathbf{bool}. \text{false}) \text{ true} \\
&\xrightarrow[\beta]{} \text{true}
\end{aligned}$$

■

By induction it could be proven that any other natural numbers must be applied $\lambda u : \mathbf{bool}. \text{false}$ to the body, making the result false, except for $\bar{0}$, where the function $f : \alpha \rightarrow \alpha$ never got applied.