

# Answers to the Exam Logical Verification

January 15, 2007

## Exercise 1.

a. A derivation showing that

$$((A \rightarrow B) \rightarrow C \rightarrow D) \rightarrow C \rightarrow B \rightarrow D$$

is a tautology:

$$[C^y] \frac{\frac{\frac{[(A \rightarrow B) \rightarrow C \rightarrow D^x] \quad \frac{[B^z]}{A \rightarrow B} I[u] \rightarrow}{C \rightarrow D} E \rightarrow}{\frac{D}{B \rightarrow D} I[z] \rightarrow} E \rightarrow}{\frac{C \rightarrow B \rightarrow D}{C \rightarrow B \rightarrow D} I[y] \rightarrow} I[x] \rightarrow}{((A \rightarrow B) \rightarrow C \rightarrow D) \rightarrow C \rightarrow B \rightarrow D} I[x] \rightarrow$$

b. We use:

$$\begin{aligned} \Gamma_0 &= \{x : (A \rightarrow B) \rightarrow (C \rightarrow D)\} \\ \Gamma_1 &= \{x : (A \rightarrow B) \rightarrow (C \rightarrow D), y : C\} \\ \Gamma_2 &= \{x : (A \rightarrow B) \rightarrow (C \rightarrow D), y : C, z : B\} \\ \Gamma_3 &= \{x : (A \rightarrow B) \rightarrow (C \rightarrow D), y : C, z : B, u : A\} \end{aligned}$$

The corresponding type derivation in simply typed  $\lambda$ -calculus:

$$\frac{\Gamma_2 \vdash y : C \quad \frac{\frac{\frac{\Gamma_2 \vdash x : (A \rightarrow B) \rightarrow (C \rightarrow D) \quad \frac{\Gamma_3 \vdash z : B}{\Gamma_2 \vdash \lambda u : A. z : A \rightarrow B}}{\Gamma_2 \vdash x(\lambda u : A. z) : C \rightarrow D}}{\Gamma_2 \vdash x(\lambda u : A. z) y : D}}{\Gamma_1 \vdash \lambda z : B. x(\lambda u : A. z) y : B \rightarrow D}}{\Gamma_0 \vdash \lambda y : C. \lambda z : B. x(\lambda u : A. z) y : C \rightarrow B \rightarrow D}}{\vdash \lambda x : (A \rightarrow B) \rightarrow (C \rightarrow D). \lambda y : C. \lambda z : B. x(\lambda u : A. z) y : ((A \rightarrow B) \rightarrow (C \rightarrow D)) \rightarrow C \rightarrow B \rightarrow D}}$$

c. The terms completed:

$$\begin{aligned} \lambda x : A \rightarrow B. \lambda y : A. (x y) \\ \lambda x : A. \lambda y : B \rightarrow C. \lambda z : B. (\lambda u : A. y) x z \\ \lambda x : A \rightarrow B. \lambda y : A. x ((\lambda z : A. y) y) \end{aligned}$$

**Exercise 2.**

- a. A proof showing that

$$(\forall x. P(x) \rightarrow A) \rightarrow (\forall y. P(y)) \rightarrow A$$

is a tautology:

$$\frac{\frac{\frac{[\forall x. P(x) \rightarrow A^u]}{P(z) \rightarrow A} E\forall}{A} E \rightarrow}{\frac{(\forall y. P(y)) \rightarrow A}{(\forall x. P(x) \rightarrow A) \rightarrow (\forall y. P(y)) \rightarrow A} I[v] \rightarrow} I[u] \rightarrow$$

- b. The corresponding  $\lambda P$ -term:

$$(\Pi x : \text{Terms. } P x \rightarrow A) \rightarrow (\Pi y : \text{Terms. } P y) \rightarrow A$$

- c. An inhabitant of

$$(\Pi x : \text{Terms. } P x \rightarrow A) \rightarrow (\Pi y : \text{Terms. } P y) \rightarrow A$$

is

$$\lambda u : (\Pi x : \text{Terms. } P x \rightarrow A). \lambda v : (\Pi y : \text{Terms. } P y). ((u z) (v z))$$

- d. No.

**Exercise 3.**

- a. A derivation in prop2 showing that  $\forall a. a \rightarrow \forall b. b \rightarrow a$  is a tautology:

$$\frac{\frac{\frac{[a^x]}{b \rightarrow a} I[y]}{\forall b. b \rightarrow a} I\forall}{\forall a. a \rightarrow \forall b. b \rightarrow a} I\forall$$

- b. The  $\lambda 2$  type corresponding to the formula  $\forall a. a \rightarrow a$ :

$$\Pi a : \star. a \rightarrow \Pi b : \star. b \rightarrow a$$

- c. A closed inhabitant of  $\Pi a : \star. a \rightarrow \Pi b : \star. b \rightarrow a$

$$\lambda a : \star. \lambda x : a. \lambda b : \star. \lambda y : b. x$$

d. Instantiation of the polymorphic identity:

$$(\lambda a : \star. \lambda x : a. x) B \rightarrow_{\beta} \lambda x : B. x$$

in  $B \rightarrow B$ .

e. Proof:

$$\frac{\frac{\frac{[a^x]}{a \rightarrow a} I[x] \rightarrow}{\forall a. a \rightarrow a} I\forall}{B \rightarrow B} E\forall$$

The part consisting of the last three lines is a detour.

**Exercise 4.**

a. We construct an inhabitant of  $A$ :

$$\frac{f : \Pi a : \star. a \quad A : \star}{f A : A}$$

b. We have

$$\lambda x : A. \lambda y : B. x : A \rightarrow B \rightarrow A$$

We use the notation  $L = \lambda x : A. \lambda y : B. x$ .

Using  $L$  we construct an inhabitant of  $A$ :

$$\frac{\frac{M : \Pi c : \star. (A \rightarrow B \rightarrow c) \rightarrow c \quad A : \star}{M A : (A \rightarrow B \rightarrow A) \rightarrow A}}{M A L : A} \quad L : A \rightarrow B \rightarrow A$$

**Exercise 5.**

a. The type inhabitation problem is:

$$\Gamma \vdash ? : A$$

Given an environment  $\Gamma$  and a type  $A$ , is there a  $M$  such that  $\Gamma \vdash M : A$ ?

b. The corresponding problem in logic is the provability problem: given a formula  $A$  and assumptions in  $\Gamma$ , is there a proof of  $A$  using only assumptions from  $\Gamma$ ?

c. The type inhabitation problem is decidable in  $\lambda \rightarrow$ .

d. The type inhabitation problem is undecidable in  $\lambda P$ .

e. The principle of program extraction is to extract an algorithm from a constructive proof. The statement that is proven usually has the form  $\forall a : A. P(x) \rightarrow \exists b : B. Q(a, b)$ .

**Exercise 6.**

- a. The type of `natlist_ind`:

```
forall P : natlist -> Prop,  
P nil ->  
(forall (n : nat) (n0 : natlist), P n0 -> P (cons n n0)) ->  
forall n : natlist, P n
```

- b. An inductive definition of polymorphic lists:

```
Inductive polylist (X : Set) : Set :=  
| polynil : polylist X  
| polycons : X -> polylist X -> polylist X.
```

- c. The type of `le 1 0` is `Prop`.

An inhabitant of `le 0 0` is `le_n 0`.

An inhabitant of `le 0 1` is `le_S 0 0 (le_n 0)`.

- d. An inductive predicate `sorted`:

```
Inductive sorted : natlist -> Prop :=  
| sorted0 : sorted nil  
| sorted1 : forall n:nat , sorted (cons n nil)  
| sorted2 : forall n h:nat , forall t:natlist ,  
  le n h -> sorted (cons h t) -> sorted (cons n (cons h t)).
```