Exam Logical Verification

January 14, 2004

There are six (6) exercises.
Answers may be given in Dutch or English. Good luck!

Exercise 1. This exercise is concerned with first-order minimal propositional logic and simply typed λ-calculus.

a. Show that the formula $((A \to B) \to C) \to B \to C$ is a tautology.
   (That is, give a proof in which all assumptions are cancelled.)
   (5 points)

b. Give the type derivation in simply typed λ-calculus corresponding to the proof of 1a.
   (5 points)

c. Replace in the following three terms the ?'s by simple types, such that we obtain typable λ-terms. (NB: it is not asked to give the type derivations.)
   $\lambda x :?. \lambda y :?. \lambda z :?. (x\, z)\, (y\, z)$
   $\lambda x :?. \lambda y :?. x\, (x\, y)$
   $\lambda x :?. \lambda y :?. \lambda z :?. (y\, x)\, (x\, z)$
   (6 points)

d. Give a proof of $A \to B \to A$ with a detour, and in which all assumptions are cancelled.
   (4 points)

Exercise 2. This exercise is concerned with first-order minimal propositional logic and simply typed λ-calculus, and consistency.

a. What is the inhabitation problem in simply typed λ-calculus?
   (3 points)

b. The inhabitation problem corresponds via the Curry-Howard-De Bruijn isomorphism to a problem in first-order minimal propositional logic.
   What problem?
   (2 points)

c. Give the correspondence between proofs in first-order minimal propositional logic and terms in simply typed λ-calculus in detail.

(5 points)

d. If a type is inhabited by a closed term, then it is inhabited by a closed term in β-normal form. A closed term in β-normal form is of the form $\lambda x : A. M$.

Does a closed inhabitant of the type $\bot$ exist? Why (not)?

(5 points)

e. Now consider $\bot$ as a formula in first-order minimal propositional logic. What can we conclude using 2abcd?

(2 points)

Exercise 3. This exercise is concerned with inductive types in Coq.

a. Give the inductive definition of the datatype `nat` of natural numbers.

(5 points)

b. Give the type of `nat_ind` which is used to give proofs by induction on the natural numbers.

(5 points)

c. Give the definition of the inductive type `bintree` of binary trees with natural numbers both on the *leaves* and on the *nodes*.

(5 points)

Exercise 4. This exercise is concerned with first-order predicate logic.

a. Give the two detours of first-order minimal predicate logic.

(5 points)

b. Show that $\forall x. (P(x) \to \neg(\forall x. \neg P(x)))$ is a tautology of first-order minimal predicate logic with $\bot$.

(5 points)

Exercise 5. This exercise is concerned with λ-calculus with dependent types (λP).

a. We assume a constructor natlist_dep that is used to build the type 'lists of natural numbers of length n' for every $n \in \{0, 1, 2, \ldots\}$.

What is the type of natlist_dep?

(3 points)

b. A typing rule that is characteristic for λP is the following:

$$\frac{\Gamma \vdash A : \mathsf{Set} \qquad \Gamma, x : A \vdash B : \mathsf{Type}}{\Gamma \vdash (x{:}A)\, B : \mathsf{Type}}$$

Explain the use of this rule in the natlist_dep example. (Hint: think of 5a.)

(5 points)

c. First-order propositional logic can be encoded in Coq using dependent types as follows:

```
(* prop representing the propositions is a Set *)
Variable prop:Set.
(* implication on prop is a binary operator *)
Variable imp: prop -> prop -> prop.
(* T expresses if a proposion in prop is valid
   if (T p) is inhabited then p is valid
   if (T p) is not inhabited then p is not valid *)
Variable T: prop -> Prop.
```

Give the types of the variables `imp_introduction` and `imp_elimination` modelling the introduction- and elimination rule of implication.

(5 points)

Exercise 6. This exercise is concerned with polymorphic λ-calculus ($\lambda 2$).

a. Define the type new_or

$$(\mathsf{new\_or}\, A\, B) = (c{:}\mathsf{Prop})\, (A \to c) \to (B \to c) \to c$$

Assume $\Gamma \vdash a : A$. Give an inhabitant of $(\mathsf{new\_or}\, A\, B)$.

(NB: it is not asked to give the type derivation.)

(5 points)

b. Assume new_or as in 6a, and in addition $\Gamma \vdash f : A \to D$, and $\Gamma \vdash g : B \to D$, and $\Gamma \vdash M : (\mathsf{new\_or}\, A\, B)$. Give an inhabitant of D.

(NB: it is not asked to give the type derivation.)

(5 points)

c. We define the booleans B and *true* (T) and *false* (F) as follows:

$\mathsf{B} = (a{:}\mathsf{Set})\, a \to a \to a$
$\mathsf{T} = \lambda a{:}\mathsf{Set}.\, \lambda x{:}a.\, \lambda y{:}a.\, x$
$\mathsf{F} = \lambda a{:}\mathsf{Set}.\, \lambda x{:}a.\, \lambda y{:}a.\, y$

Give a definition of negation in $\lambda 2$.

(5 points)

The final note is (the total amount of points plus 10) divided by 10.