Constructive Logic (15-317), Fall 2018
Assignment 3: Proof Terms, Verification and Quantification

Instructor: Karl Crary
TAs: Ankush Das, Jeanne Luning Prak, Jon Sterling
Due: Friday, September 21, 2018, 11:59 pm

This assignment must be submitted electronically at autolab. Submit your homework as a tar archive containing the following files:

- hw3.pdf (your written solutions); and
- hw3_8a.tut, hw3_8b.tut, hw3_8c.tut (your Tutch solutions for task 8)

1 All the things you can do with a ♦

Consider the ♦ connective.

\[
\frac{A \text{ true} \quad u}{B \text{ true} \quad \Diamond (A, B, C) \text{ true} \quad I_u^1} \quad \frac{A \text{ true} \quad u}{C \text{ true} \quad \Diamond (A, B, C) \text{ true} \quad I_u^2} \quad \frac{B \text{ true} \quad u}{\Diamond (A, B, C) \text{ true} \quad I_u^v} \quad \frac{C \text{ true} \quad v}{\Diamond (A, B, C) \text{ true} \quad I_v^u} \quad \frac{D \text{ true} \quad D \text{ true} \quad D \text{ true}}{\Diamond E^{u,v}}
\]

Task 1 (4 points). Invent a proof term assignment for the rules.

Task 2 (4 points). Show all the local reduction(s) and expansion(s) for these rules (proving local soundness and completeness) in proof term notation. Be sure to indicate which are reductions and which are expansions.

2 Verifications

Recall the ♦ connective from Section 1:

Task 3 (9 points). Give rules for forming the judgments that ♦(A, B, C) has a verification and that ♦(A, B, C) can be used.

Task 4 (5 points). On assignment 2, you showed

\[(\neg A \land B) \supset ((A \supset B) \supset (\neg A \supset \neg B)) \supset \bot \text{ true}.
\]

Give a verification for this proposition.

Task 5 (14 points). For each of the following propositions, give a verification and its corresponding proof term:

a. \( \bot \supset \top \)

b. \( (A \supset B) \supset (B \supset C) \supset (A \supset C) \)
3 Quantification

It is important to note that quantification extends as far to the right as syntactically possible. For example, the proposition \( \exists x : \tau.A(x) \supset \forall x : \tau.A(x) \) should be interpreted as \( \exists x : \tau.A(x) \supset (\forall x : \tau.A(x)) \) and not as \( (\exists x : \tau.A(x)) \supset (\forall x : \tau.A(x)) \). Tutch implements the same convention.

3.1 Distributivity properties

In class, we saw that universal quantification distributes over conjunction, that is,

\[
(\forall x : \tau.A(x) \land B(x)) \equiv (\forall x : \tau.A(x)) \land (\forall x : \tau.B(x) \text{ true})
\]

In this section, we will explore various other distributivity properties.

Task 6 (10 points). Dually, existential quantification distributes over disjunction, that is,

\[
(\exists x : \tau.A(x) \lor B(x)) \equiv (\exists x : \tau.A(x)) \lor (\exists x : \tau.B(x) \text{ true})
\]

In this task, you will show this equivalence by giving a natural deduction proof of each of the following directions:

a. \( (\exists x : \tau.A(x) \lor B(x)) \supset (\exists x : \tau.A(x)) \lor \exists x : \tau.B(x) \text{ true} \)

b. \( (\exists x : \tau.A(x)) \lor (\exists x : \tau.B(x)) \supset \exists x : \tau.A(x) \lor B(x) \text{ true} \)

3.2 Constructive and classical quantification

Task 7 (9 points). For each of the following judgments, give a constructive natural deduction proof and the corresponding proof term if it is constructively valid. If it is not constructively valid, state this. N.B. The following judgments are all classically valid.

a. \( (\neg \forall x : \tau.\neg A(x)) \supset \exists x : \tau.A(x) \text{ true} \)

b. \( (\exists x : \tau.A(x)) \supset (\neg \forall x : \tau.\neg A(x) \text{ true} \)

3.3 Tutch, Quantified

Tutch uses the concrete syntax \(?x:t.A(x)\) and \(!x:t.A(x)\) for \(\exists x : \tau.A(x)\) and \(\forall x : \tau.A(x)\), respectively. We encourage you to review the scoping rules for quantifiers described at the beginning of Section 3 of this assignment before starting this portion of the assignment. Please see the Tutch manual for more information on how to use quantifiers in Tutch.

Task 8 (6 points). Prove each of the following propositions using Tutch. Place the proof for part a (and only the proof for part a) in hw3_8a.tut, ..., and the proof for part c (and only the proof for part c) in hw3_8c.tut.

a. proof apply : (!x:t.A(x) => B(x))
   => (!x:t.A(x)) => (!x:t.B(x));

b. proof instance : (!x:t.A(x)) & (?y:t.B(y)) => ?z:t.A(z);

c. proof frobenius : (R & ?x:t.Q(x)) <=? x:t.(R & Q(x));