# Constructive Logic (15-317), Fall 2015
# Assignment 2: Proof Terms, Verifications, Uses

Vincent Huang (vincenth@andrew.cmu.edu)

Due Tuesday, September 22, 2015

In this homework, you will investigate how the concept of the verification and use judgements affects our logical system. You will also get some practice with proof terms, and hopefully gain some insight as to how they are assigned.

A solution for section 6 does not have to be submitted, but if you do so, you can earn up to 5 bonus points on this assignment.

The Tutch portion of your work (Section 1) should be submitted electronically using the command

```
$ /afs/andrew/course/15/317/bin/submit -r hw2 <files...>
```

from any Andrew server. You may check the status of your submission by running the command

```
$ /afs/andrew/course/15/317/bin/status hw2
```

If you have trouble running either of these commands, email Anna, Michael, or Vincent.

The written portion of your work (the remaining sections after 1) should be submitted at the beginning of class. If you are familiar with LaTeX, you are encouraged to use this document as a template for typesetting your solutions, but you may alternatively write your solutions *neatly* by hand.

## 1 Tutch Proofs

Tutch allows you to annotate your proof with proof terms by declaring it with `annotated proof`. An annotated proof is just like a regular Tutch proof, but each line `A` is annotated with the term that justifies it `M : A`.

```
annotated proof andComm : A & B => B & A =
begin
```

```
[ u : A & B;
  snd u : B;
  fst u : A;
  (snd u, fst u) : B & A];
fn u => (snd u, fst u) : A & B => B & A
end;
```

It is also possible to simply give the proof term. To give a proof term in Tutch, declare it with `term` rather than `proof`:

```
term andComm : A & B => B & A =
  fn u => (snd u, fst u);
```

For more examples, see Chapter 4 of the Tutch User's Guide. The proof terms are very similar to the ones given in lecture and are summarized in Section A.2.1 of the Guide.

**Task 1** (4 points)**.** Give annotated proofs for the following theorems using Tutch.

```
annotated proof implOr : (A => C) & (B | C) => (B => A) => C;
annotated proof loop : (A => B) => (C => A) => (C => B);
```

**Task 2** (10 points)**.** Give proof terms for the following theorems using Tutch.

```
term smap : A => (A => B) => B;
term exception : (A | B) => ~B => A;
term curry : (A & B => C) => A => B => C;
term uncurry : (A => B => C) => (A & B) => C;
term split : (A | B => C) => (A => C) & (B => C);
```

On Andrew machines, you can check your progress against the requirements file `/afs/andrew/course/15/317/req/hw2.req` by running the command

```
$ /afs/andrew/course/15/317/bin/tutch -r hw2 <files...>
```

## 2 Ups and Downs

Recall the $\Diamond$ connective introduced on the previous assignment.

$$\frac{\begin{array}{cc}\overline{A\ true}^{\,u} & \overline{B\ true}^{\,v} \\ \vdots & \vdots \\ B\ true & C\ true\end{array}}{\Diamond(A,B,C)\ true}\ \Diamond I^{u,v} \qquad \frac{\Diamond(A,B,C)\ true \quad A\ true}{B\ true}\ \Diamond E_1 \qquad \frac{\Diamond(A,B,C)\ true \quad B\ true}{C\ true}\ \Diamond E_2$$

**Task 3** (2 points)**.** Give rules using the verification ($\uparrow$) and use ($\downarrow$) judgements corresponding to these introduction and elimination rules. Note that there may be more than one correct answer.

**Task 4** (1 points)**.** Informally justify why you think the rules you provided are appropriate.

## 3 Redex Redux

**Task 5** (3 points)**.** Give *two* different natural deduction proofs of $A \wedge B \supset B \wedge A$ *true*. How many natural deduction proofs of this judgement exist? Explain clearly.

**Task 6** (3 points)**.** Give a proof of $A \wedge B \supset B \wedge A \uparrow$. How many proofs of this judgement exist? Explain clearly.

**Task 7** (3 points)**.** Give *two* different natural deduction proofs of $(A \wedge \neg A) \supset A \supset$ *A true*. How many natural deduction proofs of this judgement exist? Explain clearly.

**Task 8** (3 points)**.** Give a proof of $(A \wedge \neg A) \supset B \supset B \uparrow$. How many proofs of this judgement exist? Explain clearly.

## 4 $\diamond$ is a partscore

Refer to the rules in section 2.

**Task 9** (4 points)**.** Give a proof term assignment for the rules.

**Task 10** (3 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.

## 5 Down+B

**Task 11** (4 points)**.** Show a step-by-step reduction of the following terms, until you reach a term that cannot be further reduced. There may be more than one correct sequence of reductions. Take care to ensure your result makes sense!

(Note that case$(x, y.A, z.B)$ is short for `case x of inl y => A | inr z => B end`)

1. $(\lambda x : A \vee B.\ \text{case}(x, x.\lambda y{:}A.\ \langle x, y\rangle, x.\ \lambda y{:}A.\ \langle y, x\rangle))\ (\text{inr}\ y)$
2. $\lambda u{:}A \wedge B.\ (\lambda v{:}B \wedge A.\ \langle \text{snd}\ v, \text{fst}\ v\rangle)\ ((\lambda u{:}A \wedge B.\ \langle \text{snd}\ u, \text{fst}\ u\rangle)\ u)$

## 6 Bonus

**Task 12** (5 points)**.** Apply reduction rules to the following term until it is not possible to apply any further reduction rules.

$$(\lambda F.\ (\lambda f.\ F\,(f\,f))\ (\lambda f.\ F\,(f\,f)))\ \ (\lambda x.x)$$