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 : $A \& B \Rightarrow B \& A$

begin
\[ u : A \& B; \]
\[ \text{snd } u : B; \]
\[ \text{fst } u : A; \]
\[ (\text{snd } u, \text{fst } u) : B \& A; \] 
\[ \text{fn } u \Rightarrow (\text{snd } u, \text{fst } u) : A \& B \Rightarrow B \& A \]
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 \Rightarrow B \& A =
\[ \text{fn } u \Rightarrow (\text{snd } u, \text{fst } u); \]

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 \Rightarrow C) \& (B \mid C) \Rightarrow (B \Rightarrow A) \Rightarrow C;`
Annotated proof `loop` : `(A \Rightarrow B) \Rightarrow (C \Rightarrow A) \Rightarrow (C \Rightarrow B);`

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

`term` `smap` : `A \Rightarrow (A \Rightarrow B) \Rightarrow B;`
`term` `exception` : `(A \mid B) \Rightarrow \neg B \Rightarrow A;`
`term` `curry` : `(A \& B \Rightarrow C) \Rightarrow A \Rightarrow B \Rightarrow C;`
`term` `uncurry` : `(A \Rightarrow B \Rightarrow C) \Rightarrow (A \& B) \Rightarrow C;`
`term` `split` : `(A \mid B \Rightarrow C) \Rightarrow (A \Rightarrow C) \& (B \Rightarrow 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.

\[
\begin{array}{cccc}
A \text{ true} & B \text{ true} & v \\
\vdots & \vdots & \vdots \\
B \text{ true} & C \text{ true} & \diamond \mu, v \\
\diamond (A, B, C) \text{ true} & A \text{ true} & \diamond \mu_1 \\
B \text{ true} & \diamond (A, B, C) \text{ true} & B \text{ true} & \diamond \mu_2 \\
C \text{ true} & \diamond (A, B, C) \text{ true} & C \text{ true} & \diamond \mu_2 \\
\end{array}
\]
Task 3 (2 points). Give rules using the verification (↑) and use (↓) judgements corresponding to these introduction and elimination rules. Note that there may be more than one correct answer.

Task 4 (1 point). 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 $A \land B \supset B \land A \land A$ true. How many natural deduction proofs of this judgement exist? Explain clearly.

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

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

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

4 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 { in1 y => A } | inr z => B end)

1. $(\lambda x : A \lor B. \ \text{case}(x, x. \lambda y : A. (x, y), x. \lambda y : A. (y, x))) \ (\text{inr} \ y)$

2. $\lambda u : A \land B. (\lambda v : B \land A. \ (\text{snd} \ v, \text{fst} \ v)) ((\lambda u : A \land B. \ (\text{snd} \ u, \text{fst} \ u)) \ u)$
6 Bonus

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

\[(\lambda F. (\lambda f. F (f f)) (\lambda f. F (f f))) (\lambda x. x)\]