Midterm II Exam

15-317/657 Constructive Logic
André Platzer

November 10, 2016

Name: ______________________________

Andrew ID: ______________________________

Instructions

- This exam is closed-book with one sheet of notes permitted.
- You have 80 minutes to complete the exam.
- There are 3 problems on 6 pages.
- Read each problem carefully before attempting to solve it.
- Do not spend too much time on any one problem.
- Consider if you might want to skip a problem on a first pass and return to it later.
- Explicitly label the proof rules you use in a formal proof.
- Explain your decisions in case you run into subtleties.

	Max	Score
New Connections	90	
Quantifiers	40	
Prolog	20	
Total:	150	

1 New Connections (90 points)

Consider the new connective $\Diamond(A,B,C)$ that your friendly verificationists gave meaning to by the following introduction rule:

$$\dfrac{\begin{array}{c}\overline{A\ \mathit{true}}\ u \quad \overline{B\ \mathit{true}}\ w \\ \vdots \\ D\ \mathit{true}\end{array}}{\Diamond(A,B,D)\ \mathit{true}}\ \Diamond I^{u,w}$$

Notation: If no confusion arises, you can abbreviate $\Diamond(A,B,D)$ by $\Diamond\Diamond$ in your answers.

$\boxed{10}$ **Task 1** Give the elimination rule(s) that harmoniously fit to $\Diamond I$:

$\boxed{10}$ **Task 2** Recall the alternative notation $A_1, A_2, \ldots, A_n \vdash A$ to indicate that A *true* is provable in the natural deduction calculus from the assumptions A_1 *true* and A_2 *true* and $\ldots A_n$ *true*. Rewrite all rules for $\Diamond(A,B,D)$ in this notation $\Gamma \vdash A$.

$\boxed{10}$ **Task 3** Present corresponding sequent calculus rules for $\Diamond(A,B,D)$

You can assume without proof the usual theorems to hold after adding the $\Diamond$ connective:

Weaken: If $\Gamma \vdash C$ then $\Gamma, A \vdash C$.

Substitution: If $\Gamma \vdash A$ and $\Gamma, A \vdash C$ then $\Gamma \vdash C$.

Weakening: If $\Gamma \Longrightarrow C$ then $\Gamma, A \Longrightarrow C$.

Identity: $\Gamma, A \Longrightarrow A$.

Cut: If $\Gamma \Longrightarrow A$ and $\Gamma, A \Longrightarrow C$ then $\Gamma \Longrightarrow C$.

$\boxed{20}$ **Task 4** Prove that the sequent calculus is sound w.r.t. natural deduction, i.e., $\Gamma \Longrightarrow A$ implies $\Gamma \vdash A$, for the new cases for $\Diamond(A,B,D)$.

20 **Task 5** Prove that the sequent calculus is complete w.r.t. natural deduction, i.e., $\Gamma \vdash A$ implies $\Gamma \Longrightarrow A$, for the new cases for $\Diamond(A{,}B{,}D)$.

20 **Task 6** Prove the case of the cut theorem for sequent calculus where $\Diamond(A{,}B{,}D)$ is the principal formula in both deductions for $\Gamma \Longrightarrow \Diamond(A{,}B{,}D)$ and $\Gamma, \Diamond(A{,}B{,}D) \Longrightarrow C$ implies $\Gamma \Longrightarrow C$. Explicitly indicate why the induction hypothesis is applicable.

2 Quantifiers (40 points)

Complete the proofs of the following formulas in (either original or restricted) sequent calculus **or** explain why they are not true. Indicate the rules used or weakening, cut, identity. You can leave out typing judgments from the proofs as long as they are unambiguously correct.

20 **Task 1**

$$\frac{\ldots}{;\ \longrightarrow (\forall x{:}\tau.\ \exists y{:}\tau.\ p(x,y)) \supset (\exists y{:}\tau.\ \forall x{:}\tau.\ p(x,y))}$$

20 **Task 2**

$$\frac{\ldots}{;\ \longrightarrow (\exists y{:}\tau.\ \forall x{:}\tau.\ p(x,y)) \supset (\forall x{:}\tau.\ \exists y{:}\tau.\ p(x,y))}$$

3 Prolog (20 points)

Recall that we have studied Prolog predicates defining addition `plus/3` and multiplication `times/3` of natural numbers represented in unary from constructors `0/0` and successor `s/1`:

```
plus(N,M,R)   true iff R = N + M
times(N,M,R)  true iff R = N * M
```

10 **Task 1** Define a predicate `exp/3` for exponentiation of natural numbers in Prolog.

```
exp(B,N,R)   true iff R = (B power N)
```

10 **Task 2** Recall that modes in Prolog describe the intended ways of using a predicate. Mode `+nat` refers to an input argument of type `nat` that needs to be provided. Mode `-nat` refers to an output argument of type `nat` that will be computed by the predicate when all inputs are provided. For example the most interesting modes for `plus` and `times` were

```
plus(+nat,+nat,-nat)
plus(+nat,-nat,+nat)
times(+nat,+nat,-nat)
times(+nat,-nat,+nat)
```

Give all modes that make sense for your definition of `exp/3` and say what mathematical operation they correspond to.