Midterm II Exam

15-317/657 Constructive Logic
André Platzer

April 2, 2020

Name: ______________________________

Andrew ID: ______________________________

Instructions

- **For fairness reasons all answers must be typed on a computer in text editors/text-processing (e.g. LaTeX) and submitted as PDF. Besides PDF viewers, no other software is allowed and no handwritten answers/scans are accepted. You can use scratch paper but not hand it in.**
- **Only** the following resources can be used during this exam:
  1. 15317 lecture and recitation notes
  2. editors or text-processing software
  3. email with course staff or **private** Piazza posts with course staff

  All other communications with anyone about the exam or this course during the exam period constitute an academic integrity violation.
- You have 24 hours from when the exam was available to complete it.
- There are 3 problems on 8 pages.
- **Submit** on GradeScope → Midterm 2 → Submit assignment

	Max	Score
New Connections	70	
Theorem Proving	30	
Prolog Principles	50	
Total:	150	

1 New Connections (70 points)

The sequent calculus $\Gamma \Longrightarrow C$ came from natural deduction. Consider the new connective $@(A,B,D)$ that is given meaning by the following natural deduction introduction rule:

$$\dfrac{\dfrac{}{B\ \mathit{true}}u \quad \dfrac{}{A\ \mathit{true}}w \\ \vdots \\ D\ \mathit{true}}{@(A,B,D)\ \mathit{true}} @I^{u,w}$$

This question uses the **flat 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*.

[2] **Task 1** Rewrite the $@I^{u,w}$ natural deduction rule in its flat notation $\Gamma \vdash A$.

[5] **Task 2** Give a set of elimination rules that harmoniously fit to $@I$ (you can use flat notation):

[5] **Task 3** Give corresponding sequent calculus rules for $@(A,B,D)$ (in the original $\Longrightarrow$ calculus):

You can assume without proof the usual theorems to hold after adding the @ 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 $\Gamma \Longrightarrow C$.

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 $@(A,B,D)$.

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 $@(A,B,D)$.

18 **Task 6** Prove the case of the cut theorem for sequent calculus showing that $\Gamma \Longrightarrow C$ with principal formula $@(A,B,D)$ in the deductions for $\Gamma \Longrightarrow @(A,B,D)$ and $\Gamma, @(A,B,D) \Longrightarrow C$. Explicitly indicate why the induction hypothesis is applicable in each of its uses.

2 Theorem Proving (30 points)

Dyckhoff's contraction-free sequent calculus is *sound*, *complete*, and has the *termination* property that all its premises are strictly smaller in a well-founded ordering. Each of the following tasks considers one change to our original contraction-free sequent calculus. Concisely but clearly **explain** whether these properties still hold when replacing only the indicated rule and **mark (s)** for sound, **(u)** for unsound, **(c)** for complete, **(i)** for incomplete, **(t)** for terminating, **(n)** for nonterminating. If they fail, also give an example sequent demonstrating the failure.

To get you started here's a simple example: Replacing $\wedge R$ by $P0$ would make it

$$\frac{\Gamma \longrightarrow A \quad \Gamma \longrightarrow B}{\Gamma \longrightarrow A \wedge B}\ \wedge R \qquad \frac{\Gamma \longrightarrow A}{\Gamma \longrightarrow A \wedge B}\ P0$$

(u) because $\longrightarrow \top \wedge \bot$ proves by $P0 + \top R$ but is (constructively) false as it implies $\bot$ by $\wedge L$.
(c) every sequent provable by $\wedge R$ is provable by $P0$, which has a subset of the premises of $\wedge R$.
(t) the same ordering shows termination because $P0$ produces a subset of the premises of $\wedge R$.

10 **Task 1** What happens when we only replace rule $\bot{\supset}L$ by rule $P1$:

$$\frac{\Gamma \longrightarrow C}{\Gamma, \bot \supset B \longrightarrow C}\ \bot{\supset}L \qquad \frac{\Gamma, B \longrightarrow C}{\Gamma, \bot \supset B \longrightarrow C}\ P1$$

10 **Task 2** What happens when we only replace rule $\wedge{\supset}L$ by rule $P2$:

$$\frac{\Gamma, A_1 \supset (A_2 \supset B) \longrightarrow C}{\Gamma, (A_1 \wedge A_2) \supset B \longrightarrow C}\ \wedge{\supset}L \qquad \frac{\Gamma, (A_1 \wedge A_2) \supset B \longrightarrow C}{\Gamma, A_1 \supset (A_2 \supset B) \longrightarrow C}\ P2$$

10 **Task 3** What happens when we only replace rule ${\supset}{\supset}L$ by rule $P3$ (no need to give examples):

$$\frac{\Gamma, E \supset B, D \longrightarrow E \quad \Gamma, B \longrightarrow C}{\Gamma, (D \supset E) \supset B \longrightarrow C}\ {\supset}{\supset}L \qquad \frac{\Gamma, D \longrightarrow E \quad \Gamma, B \longrightarrow C}{\Gamma, (D \supset E) \supset B \longrightarrow C}\ P3$$

3 Prolog Principles (50 points)

This question studies ways of computing the derivative of mathematical expressions in one variable with Prolog. Assume that these mathematical expressions are represented as a data structure of type `poly` built in an arbitrary shape from these constructors:

```
add(S,T) represents the sum of S and T
mul(S,T) represents the product of S and T
x        indicates the variable (only one variable occurs so no need for a name)
n(N)     represents the number literal N (as a built-in integer)
```

In this problem you will define a predicate `diff/2` to compute the derivative of an expression represented in this way. For example, the following query is expected to succeed:

```
?- diff(add(x,n(7)), add(n(1), n(0))).
```

Modes describe the intended ways of using a predicate. Mode `+poly` indicates an input argument that needs to be provided satisfying `poly/1`. Mode `-poly` indicates an output argument satisfying `poly/1` that will be computed by the predicate when all inputs are provided.

[2] **Task 1** Write a Prolog program defining the predicate `poly/1` that simply checks whether its argument is built solely from the above constructors in any order. The `integer/1` predicate checks if its argument is an integer.

[18] **Task 2** Write a Prolog program `diff(+poly,-poly)` that takes a `poly` as an input in the first argument and produces its derivative as an output in the second argument.

10 **Task 3** Prove that your implementation from Task 2 correctly satisfies the mode `diff(+poly,-poly)`, i.e., that, for any term `T` given as input as the first argument that satisfies `poly(T)`, the goal `diff(T,D)` succeeds and computes an output term `D` that satisfies `poly(D)`.

5 **Task 4** Is the mode `diff(+poly,-poly)` for your implementation from Task 2 *total*? That is, does it always succeed at least once when given any `poly` as its first argument and a fresh variable as its second argument? Briefly explain why or why not.

5 **Task 5** Is the mode `diff(+poly,-poly)` for your implementation from Task 2 *unique*? It is unique if, for any `poly` given as the first argument, `diff/2` can never succeed with two different results for the second argument. Briefly explain why or why not.

10 **Task 6** With mode `diff(+poly,-poly)`, the predicate from Task 2 computes a derivative. For arguments satisfying `poly`, carefully explain what exactly all other modes of `diff/2` do. If they do not do anything useful clearly explain why not.