• This is a closed-book exam. No other material is permitted.
• It consists of 4 problems worth a total of 175 points.
• There are 16 pages in this exam, including 3 work sheets.
• Try to use work sheets before writing your answers. Write your answers clearly and legibly.
• You have 3 hours for this exam.

<table>
<thead>
<tr>
<th></th>
<th>Problem 1</th>
<th>Problem 2</th>
<th>Problem 3</th>
<th>Problem 4</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Max</td>
<td>30</td>
<td>55</td>
<td>50</td>
<td>40</td>
<td>175</td>
</tr>
</tbody>
</table>
1 Admissibility of Harrop’s rule [30 pts]

In constructive logic, there are some admissible rules that are not derivable. In the midterm exam, we considered Harrop’s rule as an example of such a rule and showed that it is not derivable. As we said before, in this problem, we will show that Harrop’s rule is indeed admissible. As a reminder, Harrop’s rule is as follows:

\[
\neg A \supset (B \lor C) \text{ true} \quad \text{Harrop}
\]

where \(A, B\), and \(C\) are all propositions.

You will need to use the following inference rules for neutral and normal judgments.

\[
\begin{align*}
A \downarrow x \& \vdots
B \uparrow \\
A \supset B \uparrow & \supset I_x \\
A \uparrow & \implies E \\
A \land B \uparrow & \land I \\
A \land B \downarrow & \land E_{L} \\
A \land B \downarrow & \land E_{R} \\
A \downarrow x & \\
\vdots
B \downarrow y \\
\vdots
A \uparrow & \lor L \\
A \lor B \uparrow & \lor R \\
A \lor B \uparrow & \lor E^{x,y} \\
\vdots
C \uparrow
A \downarrow x & \\
\vdots
\text{I}_x \\
\neg A \downarrow \neg E \\
A \uparrow & \bot \uparrow \\
\bot \downarrow & \neg I_x \\
\neg A \downarrow x & \\
\vdots
B \lor C \uparrow & \supset I \end{align*}
\]

- You should annotate every part of your derivation with the name of the inference rule (e.g., \(\supset I_x\)) and also a label if applicable (e.g., \(\supset I^I\)). In particular, you should annotate each hypothesis with some variable (e.g., \(A \downarrow x\)).

Since we cannot directly show that Harrop’s rule is admissible, we need to analyze the proof structure of the premise, \(\neg A \supset (B \lor C) \text{ true}\), to show that \((\neg A \supset B) \lor (\neg A \supset C) \text{ true}\) is provable. Recall that in constructive logic, \(A \text{ true}\) is provable if and only if \(A \uparrow\) is provable. Therefore we will show that Harrop’s rule is admissible by showing that \((\neg A \supset B) \lor (\neg A \supset C) \uparrow\) is provable in the presence of a proof of \(\neg A \supset (B \lor C) \uparrow\).

Proof. The proof of \(\neg A \supset (B \lor C) \uparrow\), if any, must be of the following form:

\[
\mathcal{D} \begin{cases} 
  \neg A \downarrow x \\
  \vdots \\
  B \lor C \uparrow & \supset I \\
\end{cases} \quad \neg A \supset (B \lor C) \uparrow
\]

We need to show that \((\neg A \supset B) \lor (\neg A \supset C) \uparrow\) is provable by exploring all possible cases of the structure of \(\mathcal{D}\).
Case 1. Suppose that in $D$, the rule $\neg E \downarrow$ is applied to $\neg A \downarrow^x$. Then there must be a proof of $A \uparrow$ under the hypothesis $\neg A \downarrow^x$. Let this proof be $E$. Use $E$ to build a proof of $(\neg A \supset B) \lor (\neg A \supset C) \uparrow$ (10 pts):

Case 2. Suppose that in $D$, the rule $\lor I \downarrow$ is applied to deduce $B \lor C \uparrow$. Then where must be a proof of $B \uparrow$ under the hypothesis $\neg A \downarrow^x$. Let this proof be $E$. Use $E$ to build a proof of $(\neg A \supset B) \lor (\neg A \supset C) \uparrow$ (10 pts):
Case 3. Suppose that in $\mathcal{D}$, the rule $\vee \Gamma$ is applied to deduce $B \lor C \uparrow$. Then there must be a proof of $C \uparrow$ under the hypothesis $\neg A \uparrow^\bot$. Let this proof be $\mathcal{E}$. Use $\mathcal{E}$ to build a proof of $(\neg A \supset B) \lor (\neg A \supset C) \uparrow$ (10 pts):

The above three cases are the only possible cases of the structure of $\mathcal{D}$. Therefore in the presence of a proof of $\neg A \supset (B \lor C) \uparrow$, we can always prove $(\neg A \supset B) \lor (\neg A \supset C) \uparrow$, which means that Harrop’s rule is admissible.
2 Datatypes [55 pts]

We use the following definition of proof terms for the predicate \( m < n \):

\[
\text{proof term } \quad M \ ::= \quad \cdots \mid \text{lt}_0 \mid \text{lt}_s(M) \mid \text{lt}_e(M) \mid \text{lt}_e_s(M)
\]

\[
\frac{\text{lt}_0 : 0 < s(n)}{< I_0} \quad \frac{M : m < n}{< I_s} \quad \frac{\text{lt}_e_0(M) : s(m) < s(n)}{< E_0} \quad \frac{\text{lt}_e_s(M) : m < n}{< E_s}
\]

We use the following definition of proof terms for the predicate \( m = n \):

\[
\text{proof term } \quad M \ ::= \quad \cdots \mid \text{eq}_0 \mid \text{eq}_s(M) \mid \text{eq}_e_0(M) \mid \text{eq}_e_0(M) \mid \text{eq}_e_s(M)
\]

\[
\frac{\text{eq}_0 : 0 = n}{= _{I_0}} \quad \frac{M : m = n}{= _{I_s}} \quad \frac{\text{eq}_e_0(M) : s(m) = n}{= _{E_0}} \quad \frac{\text{eq}_e_s(M) : m = n}{= _{E_s}}
\]

The following proof term represents a proof by induction on natural numbers:

\[
\text{proof term } \quad M \ ::= \quad \cdots \mid \text{ind } u(t) \text{ of } u(0) ⇒ M \mid u(s(x)) ⇒ N
\]

**Question 1. [5 pts]** Define a function \( \text{append} \) concatenating two lists:

\[
\begin{align*}
\text{append } &\text{nil} t = t \\
\text{append } &\text{cons} x l t = x :: (\text{append } l t)
\end{align*}
\]

\( \text{append} \in \text{list } τ \to \text{list } τ \to \text{list } τ \)

**Question 2. [5 pts]** Give a proof term of type \( (\exists x ∈ τ. A(x) \lor B(x)) \supset (\exists x ∈ τ. A(x)) \lor (\exists x ∈ τ. B(x)) \). You may omit type annotations in injection terms. For example, you may write \( \text{inl } M \) when \( \text{inl}_A M \) is expected.
Question 3. [5 pts] The specification of a proof term $\text{pred}$ of type $\forall x \in \text{nat}. \neg(x =_N 0) \supset \exists y \in \text{nat}. s(y) =_N x$ is:

- $x : \neg(x =_N 0)$ proof term of type $\exists y \in \text{nat}. s(y) =_N x$
- $\text{pred} \ 0 \ v : \neg(0 =_N 0) = \text{abort} \exists y \in \text{nat}. s(y) =_N 0 (v \ \text{eq} 0)$
- $\text{pred} \ s(x') \ v : \neg(s(x') =_N 0) = \langle x', \text{eq}_s(\text{eqNat} x') \rangle$

Here $\text{eqNat}$ is defined as $\lambda x \in \text{nat}. \text{ind}(u)(x) \Rightarrow \text{eq}_0(u(0)) \ | u(s(x')) \Rightarrow \text{eq}_s(u(x'))$. Give a definition of $\text{pred}$ based on this specification.

$$\text{pred} =$$

Question 4. [15 pts] We wish to find a proof term $\text{ltSucc}$ of type

$$\forall m \in \text{nat}. \forall n \in \text{nat}. m < s(n) \supset (m =_N n \lor m < n)$$

(which you used in Assignment 8). Write a specification of $\text{ltSucc}$ (similarly to $\text{pred}$) and then translate it to a definition as in Question 3.

Specification:

Definition:
Question 5. [5 pts] Write the elimination rule for the predicate \( m =_N n \) based on induction on predicates (which is given in the Course Notes):

\[
\begin{array}{c}
A(m_0, n_0) \text{ true} \\
\hline
\Rightarrow \quad =_N^{e,w,u}(m,n)
\end{array}
\]

Question 6. [5 pts] Use this elimination rule to prove \( \forall x \in \text{nat.} \forall y \in \text{nat.} x =_N y \supset y =_N x \text{ true} \):

\[
\forall x \in \text{nat.} \forall y \in \text{nat.} x =_N y \supset y =_N x \text{ true}
\]
Question 7. [5 pts] Assuming definitional equality, give a proof term of type $\forall x \in \text{nat}. \forall y \in \text{nat}. x + s(y) =_n s(x + y)$:

\[
A(t) \text{ true} \quad t =_n s
\]
\[
\frac{}{A(s) \text{ true}} \quad \text{NatEqE}
\]

Question 8. [5 pts] The following inference rule is derivable. True or false?

Question 9. [5 pts] The rule $\text{NatEqE}$ in the previous question is admissible. True or false?
3 Classical logic [50 pts]

In this problem, we use contexts $\Gamma$ and $\Delta$ defined as follows:

\[
\Gamma ::= \cdot \mid \Gamma, x : A \\
\Delta ::= \cdot \mid \Delta, x : false
\]

We use $\Gamma; \Delta \vdash K C true$ for typechecking proof terms in classical logic and $\Gamma \vdash M : C$ for typechecking proof terms in constructive logic.

Here are the rules Contra↑ and Contra↓ given in the Course Notes:

\[
\frac{\Gamma; \Delta, A false \vdash K A true}{\Gamma; \Delta \vdash K A true} \quad \text{Contra↑} \quad \frac{\Gamma; \Delta, A false \vdash K A true}{\Gamma; \Delta, A false \vdash K C true} \quad \text{Contra↓}
\]

**Question 1. [5 pts]** Use the rules Contra↑ and Contra↓ to show that the rule Peirce is derivable.

\[
\vdash \vdash_k (A \lor B) \lor A \vdash A true
\]
Question 2. [20 pts] We use the following double-negation translation for the fragment of propositional logic with implication:

\[
P^\circ = P \quad (A \supset B)^\circ = A^\circ \supset \neg\neg B^\circ
\]

For each case below, complete the CPS translation \( M^\circ \) of a given proof term \( M \). Your translation should satisfy the invariant specified in the following theorem:

**Theorem.** If \( \Gamma; \Delta \vdash K M : C \), there exists a proof term \( M^\circ \) such that \( \Gamma^\circ, \neg\Delta^\circ \vdash I M^\circ : \neg\neg C^\circ \) where

\[
\Gamma^\circ = \{ x : A^\circ \mid x : A \in \Gamma \} \\
\neg\Delta^\circ = \{ x : \neg A^\circ \mid x : A \text{ false} \in \Delta \}.
\]

Case \( x : A \in \Gamma \)
\[
\Gamma; \Delta \vdash_K x : A \quad \text{Hyp}
\]

\( x^\circ = \)

Case \( \Gamma, x : A; \Delta \vdash_K M : B \)
\[
\Gamma; \Delta \vdash_K \lambda x : A. M : A \supset B \quad \text{C}^1
\]

\( (\lambda x : A. M)^\circ = \)

Case \( \Gamma; \Delta \vdash_K M : A \supset B \quad \Gamma; \Delta \vdash_K N : A \)
\[
\Gamma; \Delta \vdash_K M N : B \quad \text{C}^E
\]

\( (M N)^\circ = \)

Case \( \Gamma; \Delta, x : A \text{ false} \vdash_K M : A \)
\[
\Gamma; \Delta \vdash_K \text{callcc} x : A \text{ false}. M : A \quad \text{Callcc}
\]

\( (\text{callcc} x : A \text{ false}. M)^\circ = \)

Case \( \Gamma; \Delta \vdash_K M : A \quad x : A \text{ false} \in \Delta \)
\[
\Gamma; \Delta \vdash_K \text{throw} M \text{ to } x : C \quad \text{Throw}
\]

\( (\text{throw } M \text{ to } x)^\circ = \)
Question 3. [25 pts] The above CPS transformation corresponds to the call-by-value reduction strategy. To be specific, the translation of $MN$ specifies that we finish evaluating $N$ before we apply the function from $M$ to the result of evaluating $N$.

In this question, we will develop a variant of the CPS translation that corresponds to the call-by-name reduction strategy. We use the following double-negation translation (known as Kolmogorov double negation translation) in which $(A ⊃ B)^\circ$ places ¬¬ before both $A^\circ$ and $B^\circ$:

$$P^\circ = P$$
$$ (A ⊃ B)^\circ = ¬¬A^\circ ⊃ ¬¬B^\circ$$

For each case below, complete the CPS translation $M^\circ$ of a given proof term $M$. Your translation should satisfy the invariant specified in the following theorem:

**Theorem.** If $\Gamma; \Delta \vdash_k M : C$, there exists a proof term $M^\circ$ such that $\Gamma^\circ, ¬\Delta^\circ \vdash_k M^\circ : ¬¬C^\circ$ where

$$\Gamma^\circ = \{ x : ¬¬A^\circ | x : A \in \Gamma \}$$
$$¬\Delta^\circ = \{ x : ¬A^\circ | x : A false \in \Delta \}.$$
4 Linear logic [40 pts]

For each proposition $A$ in linear logic below, prove its truth using natural deduction with the linear hypothetical judgment $\Delta \vdash A$. You should annotate every part of your derivation with the name of the inference rule. If its truth is unprovable, state so.

Question 1. [5 pts] $(A \otimes 1) \to A$

Question 2. [5 pts] $(A \oplus 0) \to A$

Question 3. [5 pts] $(A \& A) \to A$
Question 4. [5 pts] \((A \oplus A) \rightarrow A\)

Question 5. [5 pts] \((A \oplus (B \otimes C)) \rightarrow ((A \oplus B) \otimes (A \oplus C))\) true is provable. True or false?

Question 6. [5 pts] \((A \oplus (B \& C)) \rightarrow ((A \oplus B) \& (A \oplus C))\) true is provable. True or false?

Question 7. [5 pts] \((!A \otimes !B) \rightarrow (A \& B)\) true is provable. True or false?

Question 8. [5 pts] \(((A \rightarrow !A) \& (B \rightarrow !B)) \rightarrow ((A \otimes B) \rightarrow !(A \otimes B))\) true is provable. True or false?
Work sheet
Worksheet