15-816 Linear Logic

Midterm Examination

October 18, 2001

Name: ______________________________

- This is a closed-book exam.
- Write your answer legibly in the space provided.
- There are 12 pages in this exam, including 4 worksheets.
- It consists of 3 questions worth a total of 200 points.
- You have 80 minutes for this exam.

Problem 1	Problem 2	Problem 3	Total
60	80	60	200

1. Sequent Calculus (60 pts)

For each of the following judgments, either give a sequent calculus derivation or prove that they cannot be derived for arbitrary A and B. You may freely use theorems proved in lecture and the completeness of focusing to show that derivations are impossible.

1. (15 pts) $\cdot; !(A \& B) \Longrightarrow (!A) \otimes (!B)$

2. (15 pts) $\cdot; (!A) \otimes (!B) \Longrightarrow !(A \& B)$

3. (15 pts) $\cdot; (A \oplus B) \supset C \Longrightarrow (A \supset C) \& (B \supset C)$

4. (15 pts) $\cdot; (A \supset C) \& (B \supset C) \Longrightarrow (A \oplus B) \supset C$

2. Strict Logic (80 pts)

A *strict hypothetical judgment*

$$u_1{:}A_1\ \mathit{true}, \ldots, u_n{:}A_n\ \mathit{true} \Vdash A\ \mathit{true}$$

is a form of hypothetical judgment where each hypothesis must be used *at least once*. In this problem we are interested only in pure strict logic and do not consider adding unrestricted hypotheses.

1. (15 pts) Show the hypothesis rule.

2. (15 pts) Give the substitution property.

A *strict implication* $A \leadsto B$ is true if B is true under the strict hypothesis that A is true.

3. (10 pts) Give the introduction rule for strict implication.

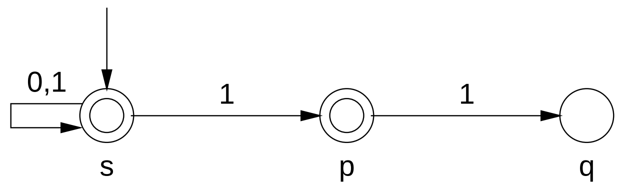
4. (10 pts) Give the elimination rule for strict implication.

5. (15 pts) Show local soundness of your rules.

6. (15 pts) Show local completeness of your rules.

3. Finite Automata (60 points)

An ordinary non-deterministic finite automaton (NFA) accepts a string iff there is a reachable accepting state. An *and-branching non-deterministic finite automaton* (&NFA) instead accepts a string iff *every* reachable state is an accepting state. For example, the following automaton accepts those strings over the alphabet $\{0, 1\}$ that do *not* end with two consecutive 1s.



1. (20 pts) Encode the &NFA above in linear logic.

2. (20 pts) Describe the encoding of &NFAs in general.

3. (20 pts) State (but do not prove) and adequacy theorem for your representation of the form:

   Given an &NFA N *and its representation* Γ *with start state* s *and accepting states* $f_1, \ldots, f_n$*. Then* N *accepts string* x *iff*

   is provable in linear logic.

Worksheet

Worksheet

Worksheet

Worksheet