CISC 465 take-home midterm

Jana Dunfield

Friday, 2022–03–04, 12:30 p.m. to Saturday, 2022–03–05, 12:30 p.m.

There is a 24-hour grace period, so the actual deadline is **Sunday** at 12:30 p.m.

Student ID number: 20311734

Please note: You must **NOT** work in a group on the midterm.

Total points: 60

1 Q1: Grammars, Inductive Definitions and Induction (15 points)

Consider the following grammar:

$$
\begin{array}{lrcl}
\text{integers} & n & & \\
\text{variables} & x, y, z & & \\
\text{terms} & M, N & ::= & x \\
 & & & \mid n \\
 & & & \mid (\lambda x.\ N) \\
 & & & \mid M(M)
\end{array}
$$

Question 1(a). This grammar corresponds to an inductive definition. Fill in the missing parts of the definition.

1. If x is a variable then x is a term.
2. If n is an integer then n is a term.
3. If x is a variable and N is a term then $(\lambda x.\ N)$ is a term.
4. If M_1 is a term and M_2 is a term then $M_1(M_2)$ is a term.

Question 1(b). For each string of symbols on the left, determine whether it is, or is not, a term (according to the above grammar).

$(\lambda x.\ z)$	is a term
$(\lambda y, y.\ y)$	is not a term
$(\lambda x.\ x(3))$	is a term
$3(x)$	is a term

A term *mentions* a variable if that variable appears somewhere in the term. For example:

- y mentions y
- $(\lambda x.\ y)$ mentions x and y
- $(\lambda z.\ z)$ mentions z
- $(\lambda x.\ 1)$ mentions x

Question 1(c). For the following conjecture, write the induction hypothesis and the appropriate set of cases to consider. You do **not** need to complete any of the cases (as it happens, the conjecture is false).

Conjecture. *For every term M, there exists a variable x such that M mentions x.*

Proof. By structural induction on M.

Induction hypothesis: for all terms M', such that $M' \prec M$, there exists a variable x such that M' mentions x.

Consider cases of M.

- Case $M = x$
- Case $M = n$
- Case $M = (\lambda x.\ N)$
- Case $M = M_1(M_2)$

$\square$

2 Q2: Natural deduction and sequent calculus (20 points)

In this question, we consider simplified versions of natural deduction (lectures 5 through 8) and sequent calculus (lecture 9).

atomic formulas	P, Q		
formulas	A, B, C ::=	P	atomic formula
		$\mid A \supset B$	implication
		$\mid A \wedge B$	conjunction (and)
contexts (in sequent calculus)	Γ ::=	$\emptyset$	empty context (no assumptions)
		$\mid \Gamma, x[A\ \mathsf{true}]$	assumption that A is true

The questions in this section involve both natural deduction and sequent calculus rules. To distinguish them, we prefix the natural deduction rules with "nat" and the sequent calculus rules with "seq". For example, the rule called $\supset$Elim in lec5-7 is called nat-$\supset$Elim here, and the rule called $\supset$Elim in lec9 is called seq-$\supset$Elim here.

$\boxed{A\ \mathsf{true}}$ A is true (in natural deduction)

$$\dfrac{\begin{array}{c} x[A\ \mathsf{true}] \\ \vdots \\ B\ \mathsf{true}\end{array}}{(A \supset B)\ \mathsf{true}}\ \text{nat-}{\supset}\text{Intro}^{x} \qquad \dfrac{A \supset B\ \mathsf{true} \qquad A\ \mathsf{true}}{B\ \mathsf{true}}\ \text{nat-}{\supset}\text{Elim}$$

$$\dfrac{A\ \mathsf{true} \qquad B\ \mathsf{true}}{A \wedge B\ \mathsf{true}}\ \text{nat-}{\wedge}\text{Intro} \qquad \dfrac{A \wedge B\ \mathsf{true}}{A\ \mathsf{true}}\ \text{nat-}{\wedge}\text{Elim1} \qquad \dfrac{A \wedge B\ \mathsf{true}}{B\ \mathsf{true}}\ \text{nat-}{\wedge}\text{Elim2}$$

Now, we give the corresponding sequent calculus rules:

$\boxed{\Gamma \vdash A\ \mathsf{true}}$ Under assumptions Γ, formula A is true (sequent calculus)

$$\dfrac{x[A\ \mathsf{true}] \in \Gamma}{\Gamma \vdash A\ \mathsf{true}}\ \text{seq-assum}$$

$$\dfrac{\Gamma, x[A\ \mathsf{true}] \vdash B\ \mathsf{true}}{\Gamma \vdash (A \supset B)\ \mathsf{true}}\ \text{seq-}{\supset}\text{Intro} \qquad \dfrac{\Gamma \vdash A \supset B\ \mathsf{true} \qquad \Gamma \vdash A\ \mathsf{true}}{\Gamma \vdash B\ \mathsf{true}}\ \text{seq-}{\supset}\text{Elim}$$

$$\dfrac{\Gamma \vdash A\ \mathsf{true} \qquad \Gamma \vdash B\ \mathsf{true}}{\Gamma \vdash A \wedge B\ \mathsf{true}}\ \text{seq-}{\wedge}\text{Intro} \qquad \dfrac{\Gamma \vdash A \wedge B\ \mathsf{true}}{\Gamma \vdash A\ \mathsf{true}}\ \text{seq-}{\wedge}\text{Elim1} \qquad \dfrac{\Gamma \vdash A \wedge B\ \mathsf{true}}{\Gamma \vdash B\ \mathsf{true}}\ \text{seq-}{\wedge}\text{Elim2}$$

Question 2(a). Complete the two derivations below.
In the second (sequent calculus) one, you may write Γ_{xy} instead of

$$\emptyset, x[Q \wedge P \text{ true}], y[Q \text{ true}]$$

Hint: When you use seq-assum, its premise should be $x[Q \wedge P \text{ true}] \in \Gamma_{xy}$.

$x[Q \wedge P \text{ true}]$
$y[Q \text{ true}]$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{Q \wedge P \text{ true}}\,x}{P \text{ true}}\,\text{nat-}\wedge\text{Elim2}}{(Q \supset P) \text{ true}}\,\text{nat-}\supset\text{Intro}^{y}}{(Q \wedge P) \supset (Q \supset P) \text{ true}}\,\text{nat-}\supset\text{Intro}^{x}$$

$\Gamma_{xy} = \emptyset, x[Q \wedge P \text{ true}], y[Q \text{ true}]$

$$\dfrac{\dfrac{\dfrac{\dfrac{x[Q \wedge P \text{ true}] \in \Gamma_{xy}}{\Gamma_{xy} \vdash Q \wedge P \text{ true}}\,\text{seq-assum}}{\Gamma_{xy} \vdash P \text{ true}}\,\text{seq-}\wedge\text{Elim2}}{x[Q \wedge P \text{ true}] \vdash (Q \supset P) \text{ true}}\,\text{seq-}\supset\text{Intro}}{\emptyset \vdash (Q \wedge P) \supset (Q \supset P) \text{ true}}\,\text{seq-}\supset\text{Intro}$$

Question 2(b). Complete the seq-$\supset$Elim case of the following proof.

Conjecture 2.1 (Soundness of sequent calculus with respect to natural deduction)**.**
If $\mathcal{D}$ derives $\Gamma \vdash A \;\mathsf{true}$ then $A \;\mathsf{true}$ under the assumptions Γ.

Proof. By induction on $\mathcal{D}$.

Induction hypothesis: If $\mathcal{D}'$ derives $\Gamma' \vdash A' \;\mathsf{true}$ and $\mathcal{D}' \prec \mathcal{D}$, then $A' \;\mathsf{true}$ under the assumptions Γ'.

Consider cases of the rule concluding $\mathcal{D}$.

- Case: seq-assum

$$\begin{array}{ll} x[A \;\mathsf{true}] \in \Gamma & \text{By inversion} \\ A \;\mathsf{true} \quad \text{under assumptions } \Gamma & \text{By assumption } x \end{array}$$

- Case: seq-$\supset$Intro

$$\begin{array}{rll} A = A_1 \supset A_2 & & \text{By inversion} \\ \Gamma \vdash A \;\mathsf{true} & & \text{Given} \\ \Gamma \vdash (A_1 \supset A_2) \;\mathsf{true} & & \text{By above equation} \\ \Gamma, x[A_1 \;\mathsf{true}] \vdash A_2 \;\mathsf{true} & & \text{By inversion} \\ & A_2 \;\mathsf{true} \quad \text{under assumptions } \Gamma, x[A_1 \;\mathsf{true}] & \text{By IH} \\ & (A_1 \supset A_2) \;\mathsf{true} \quad \text{under assumptions } \Gamma & \text{By nat-}\supset\text{Intro}^x \end{array}$$

- Case: seq-$\wedge$Intro: We should do this case, but let's not. (This means you don't have to do it.)

- Case: seq-$\wedge$Elim1

$$\begin{array}{ll} \Gamma \vdash A \;\mathsf{true} & \text{Given} \\ \Gamma \vdash A \wedge B \;\mathsf{true} & \text{By inversion} \\ \quad A \wedge B \;\mathsf{true} \quad \text{under assumptions } \Gamma & \text{By IH} \\ \quad A \;\mathsf{true} \quad \text{under assumptions } \Gamma & \text{By nat-}\wedge\text{Elim1} \end{array}$$

- Case: seq-$\wedge$Elim2

$$\begin{array}{ll} \Gamma \vdash A \;\mathsf{true} & \text{Given} \\ \Gamma \vdash A_1 \wedge A \;\mathsf{true} & \text{By inversion} \\ \quad A_1 \wedge A \;\mathsf{true} \quad \text{under assumptions } \Gamma & \text{By IH} \\ \quad A \;\mathsf{true} \quad \text{under assumptions } \Gamma & \text{By nat-}\wedge\text{Elim2} \end{array}$$

- Case: seq-$\supset$Elim

$$\begin{array}{ll} \Gamma \vdash A \;\mathsf{true} & \text{By inversion} \\ \Gamma \vdash (A_1 \supset A) \;\mathsf{true} & \text{By inversion} \\ \Gamma \vdash A_1 \;\mathsf{true} & \text{By inversion} \\ \quad (A_1 \supset A) \;\mathsf{true} \quad \text{under assumptions } \Gamma & \text{By IH} \\ \quad A_1 \;\mathsf{true} \quad \text{under assumptions } \Gamma & \text{By IH} \\ \quad A \;\mathsf{true} \quad \text{under assumptions } \Gamma & \text{By nat-}\supset\text{Elim} \end{array}$$

□

3 Q3: Choice (25 points)

Consider the language of terms from Q1, extended with a *choice operator* $\mathsf{choose}(M_1,\ M_2)$, which uses *either* M_1 or M_2, unpredictably. The new grammar is:

$$
\begin{array}{lrcll}
\text{integers} & n & & & \\
\text{variables} & x, y, z & & & \\
\text{values} & v & ::= & x \mid (\lambda x.\ N) & \\
\text{(Q3) terms} & M, N & ::= & x & \text{variable} \\
 & & & \mid n & \text{integer} \\
 & & & \mid (\lambda x.\ N) & \text{abstraction} \\
 & & & \mid M(M) & \text{application} \\
 & & & \mid \mathsf{choose}(M,\ M) & \text{choice}
\end{array}
$$

The rules for the choice operator are eval-choose1 and eval-choose2. The other rules are somewhat similar to those in Lecture 10, with different syntax. The rule eval-app corresponds to the rule eval-call in Lecture 10.

$\boxed{M \Downarrow v}$ Term M evaluates to v

$$
\frac{}{v \Downarrow v}\ \text{eval-value}
\qquad
\frac{M_1 \Downarrow (\lambda x.\ N) \qquad M_2 \Downarrow v_2 \qquad [v_2/x]N \Downarrow v}{M_1(M_2) \Downarrow v}\ \text{eval-app}
$$

$$
\frac{M_1 \Downarrow v_1}{\mathsf{choose}(M_1,\ M_2) \Downarrow v_1}\ \text{eval-choose1}
\qquad
\frac{M_2 \Downarrow v_2}{\mathsf{choose}(M_1,\ M_2) \Downarrow v_2}\ \text{eval-choose2}
$$

Give a counterexample to the following conjecture:

Conjecture 3.1. *If* $M \Downarrow v$ *and* $N \Downarrow v$ *then* $M = N$.

Counterexample:

$M = \mathsf{choose}(3,\ 5)$

$N = \mathsf{choose}(3,\ 3)$

$v = 3$

Question 3(b). A useful property of (some) programming languages is *type preservation*: if a program gives an answer, the answer should have the same type as the program. For example, if $\mathsf{choose}(1,\ 2)$ has type int, and $\mathsf{choose}(1,\ 2) \Downarrow v$, then v should also have type int.

Giving a complete set of typing rules would take us past the lecture notes covered in class so far. However, this question uses a drastically simplified type system, with only two types: int, the type of integers (like n), and $\mathsf{int} \to \mathsf{int}$, the type of the identity function $(\lambda x.\ x)$ (which corresponds to `(Lam x x)` in the language of lec10.pdf).

types	S, T ::=	$\mathsf{int} \mid \mathsf{int} \to \mathsf{int}$	
typing contexts	Γ ::=	$\emptyset$	empty context
		$\mid \Gamma, x : S$	x has type S

The typing rules for this question are:

$\boxed{\Gamma \vdash M : S}$ term M has type S

$$\frac{}{\Gamma \vdash n : \mathsf{int}}\ \text{type-int} \qquad \frac{}{\Gamma \vdash (\lambda x.\ x) : \mathsf{int} \to \mathsf{int}}\ \text{type-identity} \qquad \frac{\Gamma \vdash M_1 : S \to T \qquad \Gamma \vdash M_2 : S}{\Gamma \vdash M_1(M_2) : T}\ \text{type-app}$$

$$\frac{\Gamma \vdash M_1 : S \qquad \Gamma \vdash M_2 : S}{\Gamma \vdash \mathsf{choose}(M_1,\ M_2) : S}\ \text{type-choose}$$

There are many, many terms that should be typable but aren't, such as $(\lambda x.\ 3)$, but we ignore them in this question.

3(b) part 1. Complete the following derivations:

$$\frac{\dfrac{}{\emptyset \vdash 4 : \mathsf{int}}\ \text{type-int} \qquad \dfrac{}{\emptyset \vdash 5 : \mathsf{int}}\ \text{type-int}}{\emptyset \vdash \mathsf{choose}(4,\ 5) : \mathsf{int}}\ \text{type-choose}$$

$$\frac{\dfrac{}{\emptyset \vdash (\lambda x.\ x) : \mathsf{int} \to \mathsf{int}}\ \text{type-identity} \qquad \dfrac{}{\emptyset \vdash (\lambda y.\ y) : \mathsf{int} \to \mathsf{int}}\ \text{type-identity}}{\emptyset \vdash \mathsf{choose}((\lambda x.\ x),\ (\lambda y.\ y)) : \mathsf{int} \to \mathsf{int}}\ \text{type-choose}$$

3(b) part 2. Complete one case of type preservation: the case for eval-choose2.

Proving some of the other cases would require additional lemmas, such as a substitution lemma, but the case for eval-choose2 should not need any lemmas.

Conjecture 3.2. *If* $\emptyset \vdash M : S$ *and* $\mathcal{D}$ *derives* $M \Downarrow v$ *then* $\emptyset \vdash v : S$.

Proof. By induction on the derivation $\mathcal{D}$.

The induction hypothesis is: If $\mathcal{D}' \prec \mathcal{D}$ and $\emptyset \vdash M' : S'$ and $\mathcal{D}'$ derives $M' \Downarrow v'$ then $\emptyset \vdash v' : S'$.

Consider cases of the rule concluding $\mathcal{D}$.

- Case for eval-choose2:

$\emptyset \vdash M_2 : S$	By inversion on type-choose
$M_2 \Downarrow v$	By inversion on eval-choose2
$\emptyset \vdash v : S$	By IH

□

3(b) part 3. The statement of type preservation contains several objects that we could potentially induct on. I chose to induct on the derivation of $M \Downarrow v$. Considering only the eval-choose2 case, would inducting on S have worked? That is, the IH would require $S' \prec S$, instead of $\mathcal{D}' \prec \mathcal{D}$. (For example, $\mathsf{int} \prec \mathsf{int} \rightarrow \mathsf{int}$, but $\mathsf{int} \rightarrow \mathsf{int} \not\prec \mathsf{int} \rightarrow \mathsf{int}$.)

No. The S of the premises of type-choose is the same as the S of the conclusion of type-choose, so S does not get smaller from its conclusion to any of its premises.