CMSC 22100/32100: Programming Languages

Homework 1

M. Blume **Due: October 7, 2008**

---

1. Consider the rules:

$$\frac{}{\textbf{zero nat}}\ \textsc{zero} \qquad \frac{n\ \textbf{nat}}{\textbf{succ}(n)\ \textbf{nat}}\ \textsc{succ} \qquad \frac{}{\textbf{nil list}}\ \textsc{nil} \qquad \frac{n\ \textbf{nat} \qquad l\ \textbf{list}}{\textbf{cons}(n,l)\ \textbf{list}}\ \textsc{cons}$$

These rules define a set of terms **nat** representing natural numbers in Peano encoding and a set of terms **list** representing lists of such numbers.

We can inductively (*i.e.*, recursively) define the following *append* function on lists:

$$\begin{aligned} \mathit{append}(\textbf{nil}, m) &= m \\ \mathit{append}(\textbf{cons}(n,l), m) &= \textbf{cons}(n, \mathit{append}(l,m)) \end{aligned}$$

(a) Represent *append* as a ternary relation and give its definition inductively.
9pt

(b) Write down a set of inference rules that defines the same ternary relation.
10pt

(c) Prove that the so-defined relation is single-valued, *i.e.*, that it represents a binary function.
12pt

2. (See Chapter 2.1) Let $s \mapsto s'$ be some arbitrary binary relation and let $\mapsto^*$ be defined by the following two inference rules:

$$\frac{}{s \mapsto^* s}\ \textsc{refl} \qquad\qquad \frac{s \mapsto s' \qquad s' \mapsto^* s''}{s \mapsto^* s''}\ \textsc{trans}$$

Prove that $\mapsto^*$ is indeed transitive, *i.e.*, that $\forall s, s', s''.\ s \mapsto^* s' \wedge s' \mapsto^* s'' \Rightarrow s \mapsto^* s''$.
20pt

3. Consider a language where all values are Peano-encoded natural numbers given by the **nat** judgment from question 1. The expressions e of the language shall be of one of the following forms: **zero** representing the constant 0, **succ**(e) representing the operation of producing the successor of a given argument, **pred**(e) representing the operation of producing the

natural predecessor [1] of a given argument, and $\mathbf{if0}(e_1, e_2, e_3)$ representing a tests of e_1 for being 0, returning the result of e_2 if it is or the result of e_3 if it is not.

6pt (a) Give a definition of e in BNF style.

(b) Give equivalent inference rules for a judgment e **exp** which holds if
8pt e is an expression of the language.

(c) Give a set of inference rules for judgments of the form $e \Rightarrow n$ where e is an expression and n is a natural number (in Peano-encoding). The judgment should express the "evaluates-to" relation in the style of a big-step operational semantics and must correspond to the informal
10pt description given above.

10pt (d) Prove that if $e \Rightarrow n$ is derivable, then so is e **exp** as well as n **nat**.

15pt (e) Prove that the relation $\Rightarrow$ defined by your rules is single-valued.

[1] The natural predecessor of $n + 1$ is n, and the natural predecessor of 0 is taken to be 0.