Operational and Denotational Semantics

Please turn in by April 5, 2011 (-1 point penalty per day late)

We consider an enriched, typed λ-calculus with constants for arithmetic expressions and recursion. Note that the essential question is question 9. This is the one that takes the longest to answer, and the one that will really tell me whether you understand (variants of) techniques we have seen in the λ-calculus.

The *types* are $\sigma, \tau ::= \texttt{Nat} \mid \sigma \to \tau$. All the terms in the language come with explicit types. In particular, we assume an infinite, countable set of variables of each type τ, and write x_τ for a variable of type τ.

The terms t of type τ (in short, $t : \tau$) are defined by induction on their size by :

- every variable x_τ is of type τ ;
- if $N : \tau$, then $\lambda x_\sigma \cdot N$ is a term of type $\sigma \to \tau$;
- if $M : \sigma \to \tau$ and $N : \sigma$, then MN is a term of type τ ;
- for each $n \in \mathbb{N}$, there is a distinct constant $\underline{n} : \texttt{Nat}$;
- there are distinct constants $\texttt{pred} : \texttt{Nat} \to \texttt{Nat}$ (subtract one), $\texttt{succ} : \texttt{Nat}$ (add one), $\texttt{ifz}_\tau : \texttt{Nat} \to \tau \to \tau \to \tau$ (test if first argument equals 0), and $Y_\tau : (\tau \to \tau) \to \tau$ (fixpoint, recursion).

As in the λ-calculus, the terms are understood up to α-renaming. This takes the special form that x_σ can be replaced by any fresh variable y_σ of the *same* type σ in $\lambda x_\sigma \cdot t$. We drop type indices whenever they are irrelevant or can be reconstructed from context.

We define its semantics not through reduction, but by building a machine directly.

The *contexts* are defined by the grammar $E := \_ \mid EN \mid \texttt{succ}\, E \mid \texttt{pred}\, E \mid \texttt{ifz}\ E\ N\ P$, where N, P denote terms. . We see contexts as particular terms, with a unique occurrence of a specific variable $\_$ that does not occur in any term, called the *hole*. A context E is of type $\sigma \vdash \tau$ when E is of type τ, assuming the hole $\_$ of type σ. $E[M]$ denotes the replacement of the hole by M (of type σ).

Our machine is a transition system whose configurations are pairs $E \cdot M$. Intuitively, in such a configuration, the machine is in the process of evaluating $E[M]$, and the focus in currently on the subterm M.

The machine rules come into two groups. The first form the *redex discovery* rules. E.g., in $(DApp)$, the machine tries to evaluate MN, and proceeds by pushing the argument N into the

context and focusing on the function part M.

$$\begin{array}{cccc} (DApp) & E \cdot MN \to E[\_N] \cdot M & (DIf) & E[\_MNP] \cdot \texttt{ifz} \to E[\texttt{ifz}\ \_\ N\ P] \cdot M \\ (DPred) & E[\_N] \cdot \texttt{pred} \to E[\texttt{pred}\ \_] \cdot N & (DSucc) & E[\_N] \cdot \texttt{succ} \to E[\texttt{succ}\ \_] \cdot N \end{array}$$

The second group of rules *computes* :

$$\begin{array}{cccc} (\beta) & E[\_N] \cdot \lambda x \cdot P \to E \cdot P[x := N] & (Y) & E[\_N] \cdot Y \to E \cdot N(YN) \\ (\texttt{pred}) & E[\texttt{pred}\ \_] \cdot \underline{n+1} \to E \cdot \underline{n} & (\texttt{succ}) & E[\texttt{succ}\ \_] \cdot \underline{n} \to E \cdot \underline{n+1} \\ (\texttt{ifz}0) & E[\texttt{ifz}\ \_\ N\ P] \cdot \underline{0} \to E \cdot N & (\texttt{ifz}1) & E[\texttt{ifz}\ \_\ N\ P] \cdot \underline{n+1} \to E \cdot P \end{array}$$

We do *not* take the closure of $\to$ under contexts, whatever this may mean. The relation $\to$ is entirely specified by the rules above.

We also consider a *denotational semantics* of the above terms. First, we define the cpo $[\![\tau]\!]$ of all *values* of type τ :

- $[\![\texttt{Nat}]\!]$ is $\mathbb{N}_\perp$, the set of all natural numbers plus an added, so-called *bottom* element $\perp$. These are ordered by $m \leq n$ iff $m = \perp$ or $m = n$. (Think as all element of $\mathbb{N}$ being incomparable and above $\perp$.)
- $[\![\sigma \to \tau]\!]$ is the dcpo $[[\![\sigma]\!] \to [\![\tau]\!]]$ of all continuous maps from $[\![\sigma]\!]$ to $[\![\tau]\!]$. They are ordered by $f \leq g$ iff $f(x) \leq g(x)$ in $[\![\tau]\!]$ for every $x \in [\![\sigma]\!]$.

1. Show that every term t has exactly one type.
2. Show that $[\![\tau]\!]$ has a least element $\perp_\tau$, for every type τ. By abuse of language, we shall write $\perp$ instead of $\perp_\tau$, and call it "bottom". It is useful to think of $\perp$ as non-termination.

Then we define the semantics $[\![t]\!]\,\rho$ of terms $t : \tau$ as values in $[\![\tau]\!]$ in *environments* ρ mapping each variable x_σ to an element of $[\![\sigma]\!]$, by :

- $[\![x_\sigma]\!]\,\rho = \rho(x_\sigma)$;
- $[\![MN]\!]\,\rho = [\![M]\!]\,\rho([\![N]\!]\,\rho)$;
- $[\![\lambda x_\sigma \cdot N]\!]\,\rho$ is the function that maps each $v \in [\![\sigma]\!]$ to $[\![N]\!]\,(\rho[x_\sigma := v])$;
- $[\![\underline{n}]\!]\,\rho = n$, for every $n \in \mathbb{N}$;
- $[\![\texttt{pred}]\!]\,\rho$ is the map that sends $n+1$ to $n \in \mathbb{N}$, and 0 and $\perp$ to $\perp$;
- $[\![\texttt{succ}]\!]\,\rho$ is the map that sends $n \in \mathbb{N}$ to $n+1$, and $\perp$ to $\perp$;
- $[\![\texttt{ifz}_\tau]\!]\,\rho$ is the map that sends u, v, w to $\perp$ if $u = \perp$, to v if $u = 0$, and to w if $u \neq 0, \perp$;
- $[\![Y_\tau]\!]\,\rho$ is the map that sends $f \in [[\![\tau]\!] \to [\![\tau]\!]]$ to $\sup_{n\in\mathbb{N}} f^n(\perp_\tau)$.

3. Given any continuous map $f \in [[\![\tau]\!] \to [\![\tau]\!]]$, show that $lfp(f) = \sup_{n\in\mathbb{N}} f^n(\perp_\tau)$ exists in $[\![\tau]\!]$, and is the *least fixpoint* of f. A fixpoint of f is an element v such that $f(v) = v$. It is least iff $v \leq w$ for every other fixpoint w. (This is meant to explain the definition of $[\![Y_\tau]\!]\,\rho$.)
4. Establish *soundness* : if $E \cdot M \to^* \_ \cdot \underline{n}$, then $[\![E[M]]\!]\,\rho = n$, for every $n \in \mathbb{N}$, and every environment ρ.

The rest of the questions aim at establishing the converse of soundness, a property known as *computational adequacy*. This is harder : notice in particular that computational adequacy entails that if $[\![E[M]]\!]\,\rho = n$ for every $n \in \mathbb{N}$, and every environment ρ, then $E \cdot M$ must terminate (on the configuration $\_ \cdot \underline{n}$).

We first show that we can only hope to prove computational adequacy in a limited setting. First, we only consider *ground configurations* : call a term *ground* iff it has no free variable, and a configuration $E \cdot M$ *ground* iff $E[M]$ is ground. Then $[\![E[M]]\!]\,\rho$ does not depend on ρ, and we shall write it $[\![E[M]]\!]$.

5. One might think of proving computational adequacy at every type τ, i.e., that if $E \cdot M$ is ground, and $[\![E[M]]\!]$ is a value $v \in [\![\tau]\!]$ (the same for every environment ρ), then $E \cdot M \to^* \_ \cdot M'$ for some canonical ground term $M' : \tau$ with $[\![M']\!] = v$. By "canonical", we mean that there is a unique canonical term M' such that $[\![M']\!] = v$. Show that this is hopeless : canonicality fails, at least for some type τ other than $\mathtt{Nat}$.

Define $\precsim_\sigma$ on ground terms by $M \precsim_\sigma N$ iff for every context E of type $\sigma \vdash \mathtt{Nat}$, if $E \cdot M \to^* \_ \cdot \underline{n}$ then $E \cdot N \to^* \_ \cdot \underline{n}$.

Extend $\precsim_\sigma$ to non-ground terms by $M \precsim_\sigma N$ iff $M\theta \precsim_\sigma N\theta$ for every well-typed substitution θ of ground terms for all variables in M and N. A substitution θ is *well-typed* iff $x_\sigma\theta$ is a term of type σ for every variable x_σ in its domain.

6. Show that :
   - $M[x_\sigma := N] \precsim_\tau (\lambda x_\sigma \cdot M)N$ whenever $M : \tau$, $N : \sigma$;
   - $M(YM) \precsim_\tau YM$ provided $M : \tau \to \tau$;
   - if $\underline{n} \precsim_{\mathtt{Nat}} M$ then $\underline{n+1} \precsim_{\mathtt{Nat}} \mathtt{succ}\, M$;
   - if $\underline{n+1} \precsim_{\mathtt{Nat}} M$ then $\underline{n} \precsim_{\mathtt{Nat}} \mathtt{pred}\, M$;
   - if $\underline{0} \precsim_{\mathtt{Nat}} M$ then $N \precsim_\tau \mathtt{ifz}\ M\ N\ P$ (where $N : \tau$, $P : \tau$),
   - if $\underline{n+1} \precsim_{\mathtt{Nat}} M$ then $P \precsim_\tau \mathtt{ifz}\ M\ N\ P$ (where $N : \tau$, $P : \tau$).

The main tool in the proof of computational adequacy is the use of a so-called *logical relation*. This is a notion similar to the reducibility sets RED_τ defined in the lectures, except there are now binary relations, i.e., sets of pairs.

Here, a logical relation is a family of binary relations R_τ, indexed by types τ, between values in $[\![\tau]\!]$ and ground terms $M : \tau$. These are defined by induction on types as follows. For short, we say "for all $N\ R_\sigma\ v$" instead of "for every term $N : \sigma$, every value $v \in [\![\sigma]\!]$, if $N\ R_\sigma\ v$ then".

- $M\ R_{\mathtt{Nat}}\ u$ iff $u = \bot$, or u is a natural number n and $\underline{n} \precsim_{\mathtt{Nat}} M$.
- $M\ R_{\sigma\to\tau}\ f$ iff for all $N\ R_\sigma\ v$, $MN\ R_\tau\ f(v)$.

7. Given $M : \tau$, let MR_τ be the set $\{u \in [\![\tau]\!] \mid M\ R_\tau\ u\}$. Show that MR_τ is *Scott-closed*, i.e., both *downward closed* (if $u \leq v$ and $M\ R_\tau\ v$ then $M\ R_\tau\ u$) and stable under directed sups (if $(u_i)_{i\in I}$ is a directed family in $[\![\tau]\!]$, and $M\ R_\tau\ u_i$ for every $i \in I$, then $M\ R_\tau \sup_{i\in I} u_i$). Show also that MR_τ is non-empty, i.e., contains $\bot$.
8. Show that if $M \precsim_\tau N$ and $M\ R_\tau\ u$, then $N\ R_\tau\ u$. In other words, the set $R_\tau u = \{M : \tau \mid M\ R_\tau\ u\}$ is *upward closed* in $\precsim_\tau$.
9. Given a well-typed substitution θ, and an environment ρ, write $\theta\ R_*\ \rho$ iff $x_\sigma\theta\ R_\sigma\ \rho(x_\sigma)$ for every variable x_σ in the domain of θ.

   Prove the *Basic Lemma* : if $\theta\ R_*\ \rho$, and M is a term of type τ, then $M\theta\ R_\tau\ [\![M]\!]\,\rho$.
10. Conclude that *computational adequacy* holds : given a ground configuration $E \cdot M$ such that $E[M] : \mathtt{Nat}$, $[\![E[M]]\!] = n \in \mathbb{N}$ if and only if $E \cdot M \to^* \_ \cdot \underline{n}$.