Computer Science 190 (Autumn Term, 2003)

Programming Language Theory

Problem Set 4/Take-home examination

(Unlike previous problem sets, this set of problems is to be solved without *consulting others in the class. You may however ask questions of the course instructor or teaching assistants.)*

Your solutions are due on **Thursday, December 11 at 10am,** in TA Jessica Littman's mailbox in the computer science department office. At that exact hour, Jess is going to sweep into the office, pick them up, and leave. *If you have to leave town earlier than that,* you should mail them to her in the office.

Problem 0. We showed in class lecture that every simply-typed λ-term E has a normal form. In this exercise, we want to prove similar upper bounds on the *length* of the normal form for a restricted reduction strategy. We consider a calculus where we have only variables, application, and λ-abstraction—no pairing, case statements, addition, etc.

Given *any* λ-term T, we define a *complete development* of T as follows. *Underline* every β-redex in T. Now begin reduction; as you replace subexpressions $\underline{(\lambda x.E)F}$ by $E[F/x]$, you may possibly create *new* redexes that are *not* underlined, as well as create new redexes that are *copies* of old, underlined ones. A complete reduction *only* reduces underlined redexes. (Obviously, when $\underline{(\lambda x.E)F}$ reduces to $E[F/x]$, the underlining is lost.)

For example, $\underline{(\lambda x.x(xy))(\lambda z.z)}$ reduces to $(\lambda z.z)((\lambda z.z)y)$, creating two new redexes, but neither is underlined, so they are not reduced in a complete development. On the other hand, $\underline{(\lambda x.x(xy))\underline{((\lambda w.w)(\lambda z.z))}}$ reduces to $\underline{((\lambda w.w)(\lambda z.z))}(\underline{((\lambda w.w)(\lambda z.z))}y)$, and the two copies *are* reduced in a complete development.

[Part (a)] Prove that a complete development is finite—namely, show that for any λ-term with underlined redexes, there is always a finite set of reductions (a *reduction strategy*) which produces another λ-term without any underlined redexes.

[Part (b)] Define the *length* $|T|$ of a λ-term T as the number of bits, or symbols it takes to write T down. If T' is a complete development of T, prove that $|T'| \leq 2^{|T|}$. (You may alternatively prove this theorem with 2 replaced by your favorite constant, should you prefer.)

[Part (c)] **(Difficult.)** Consider simple types over a single base type o, and define the *order* of a type as $\text{order}(o) = 0$, and $\text{order}(\alpha \to \beta) = \max\{1 + \text{order}(\alpha), \text{order}(\beta)\}$. Compute the order of $o, o \to o$, and $(o \to o) \to (o \to o)$. Now define $\mathbf{K}(t,n)$ on integers as $\mathbf{K}(0,n)$ and $\mathbf{K}(t+1,n) = 2^{\mathbf{K}(t,n)}$. Compute $\mathbf{K}(i,2)$ for $i = 0,1,2,3,4$. Then show (this is the hard part!) that if T is a simply-typed term with type α, then its *normal form* has length $O(\mathbf{K}(\text{order}(\alpha), |T|))$.

Problem 1. Let $\omega = \{0, 1, 2, \ldots\}$ and $\mathcal{P}\omega = \{S \mid S \subseteq \omega\}$, so $\mathcal{P}\omega$ is the set of all subsets of integers ω. Observe that $\mathcal{P}\omega$ is a CPO, where $\bigvee$ is $\bigcup$ (union) and $\leq$ is $\subseteq$ (subset)—if $\mathbf{T} \subseteq \mathcal{P}\omega$ ($\mathbf{T}$ is now a set of subsets of integers), then $\bigcup \mathbf{T}$ exists, *whether or not* S is directed! Note that $\bigcup \mathbf{T} = \bigcup\{S \mid S \in \mathbf{T}\}$—the least upper bound of a set $\mathbf{T}$ of sets is just a set containing exactly the elements of each $S \in \mathbf{T}$.

Now let $m(S) = \min\{i \in \omega \mid i \notin S\}$, and define $f : \mathcal{P}\omega \to \mathcal{P}\omega$ to be $f(S) = S \cup \{m(S)\}$. So, for example $f(\{0, 1, 2, 4, 7\}) = \{0, 1, 2, 3, 4, 7\}$. We want to prove that f is a *continuous* function: that is, for any *directed* set $\mathbf{T}$ of subsets of ω, $f(\bigcup \mathbf{T}) = \bigcup f(\mathbf{T})$, which is just shorthand for

$$f(\bigcup\{S \mid S \in \mathbf{T}\}) = \bigcup\{f(S) \mid S \in \mathbf{T}\}.$$

Each part below has a short answer: put them together, and you have a proof that f is continuous. If you don't use the fact that $\mathbf{T}$ is directed, *somewhere* in the proof, you've made a mistake.

[Part (a)] Show that if $\mathbf{T}$ is *not* directed, then $f(\bigcup \mathbf{T}) = \bigcup f(\mathbf{T})$ is not necessarily true. That is, *find a counterexample:* you should be able to do this with $\mathbf{T}$ having 2 elements, each a small (finite!) set of integers.

[Part (b)] Prove that if $A \subseteq B$, then $m(A) \leq m(B)$ and $f(A) \subseteq f(B)$.

[Part (c)] Prove that $\bigcup f(\mathbf{T}) \subseteq f(\bigcup \mathbf{T})$.

[Part (d)] Prove that for every $S \in \mathbf{T}$, $m(S) \leq m(\bigcup \mathbf{T})$.

[Part (e)] Let $k = m(\bigcup \mathbf{T})$; prove that for each $j = 0, 1, \ldots, k-1$, there exists some $S \in \mathbf{T}$ where $j \in S$ (call this set S_j as a reminder that it contains j).

[Part (f)] Prove that there exists a set $D \in \mathbf{T}$ where $S_j \subseteq D$ for each $j = 0, 1, \ldots, k-1$.

[Part (g)] Prove that $m(D) \geq k$; then prove $m(D) = k$.

[Part (h)] Prove that $f(\bigcup \mathbf{T}) \subseteq \bigcup f(\mathbf{T})$.

Problem 2. Recall our definition of the factorial function, $\mathit{fact} = \bigvee\{F^n(\bot) \mid n \geq 0\}$ of type $\mathit{nat} \to \mathit{nat}$, where

$$F = \lambda f.\lambda n.\text{if } n = 0 \text{ then } 1 \text{ else } n \times f(n-1)$$

[Part (a)] Give an alternative characterization of *fact* as the $\bigvee$ of a set of step functions $a \searrow b$—specifically, what are the step functions?

[Part (b)] Give a characterization of $F^n(\bot)$ as the $\bigvee$ of a set of step functions $a \searrow b$. What are the step functions?

Problem 3. In a class warmup to prove the Full Abstraction Theorem, we showed that if a is a compact element of a PCPO A, then for any element b of a PCPO B, the function $a \searrow b$ is a continuous function in the PCPO $\mathbf{A} \to \mathbf{B}$. Show in contrast that if a is *not* compact, then $a \searrow b$ is not necessarily a continuous function. (In other words, find a counterexample.)

Problem 4. Assume that the meaning of every PCF-definable element of type $A \to B$ can be written as $\bigvee\{a_i \searrow b_i \mid i \in I\}$, for a_i (b_i) an element in the set denoted by A (B). Define a suitable application function $\text{App}^{A,B}$ and show it is continuous.

Problem 5. Consider the full type frame over the base type $A^o = \{0, 1\}$. It is easy to see that there are four elements of $A^{o \to o}$ and $4^4 = 256$ elements of $A^{(o \to o) \to (o \to o)}$.

[Part (a)] Calculate the meaning of the term $\mathbf{2} = \lambda s : o \to o.\lambda z : o.s(sz)$ in this model using the inductive definition of meaning.

[Part (b)] Describe the four elements of $A^{o \to o}$ and state the result of applying the function defined by $\mathbf{2}$ to each of them.

[Part (c)] Recall that the meaning of any *typed* λ*-term* is the meaning of its unique *normal form*. How many elements in each of the type sets $A^o, A^{o \to o}, A^{(o \to o) \to (o \to o)}$ are the meanings of *closed* λ-terms without term constants?

Problem 6. An alternate definition of complete partial orders uses *chains* instead of *directed sets*. Let $\mathcal{D} = \langle D, \leq \rangle$ be a *countable* set (read that again—*countable*) and let $\leq$ be a partial order on D. A *chain* is a countably infinite set $d_0 \leq d_1 \leq d_2 \leq \cdots$ of elements of D ordered by $\leq$. If $\mathcal{D}$ is a *chain-CPO* when every chain has a least upper bound, prove $\mathcal{D}$ is a chain-CPO if and only if it is a CPO (where every *directed set* must have a least upper bound). *Hint:* In the "only if" direction, where we assume $\mathcal{D}$ is a chain-CPO, construct from every directed set S (which must be countable) a chain whose least upper bound must be the least upper bound of S.

Problem 7. Using the approximation theorem (Theorem 5.4.6 in the textbook), prove that the inference rule used in Exercise 2.3.3 is sound for the standard model. Specifically, show that if $M \twoheadrightarrow NM$, then $\bigvee \mathit{anf}(M) = \bigvee \mathit{anf}(\mathrm{fix} N)$.