Advanced Type Systems
Homework #4

Instructor: Derek Dreyer

Assigned: February 15, 2006
Due: February 27, 2006

1 Termination in the Presence of Existentials

Prove that call-by-value System F is still terminating even when extended with existential types. This result should not be surprising, since it is well-known that existential types are encodable in System F in terms of universal types (see Section 11.3.5 in Girard's *Proofs and Types*).

Here are the appropriate extensions:

$$\begin{array}{lrcl} \text{Types} & \sigma, \tau & ::= & \cdots \mid \exists \alpha.\, \tau \\ \text{Terms} & e, f & ::= & \cdots \mid \texttt{pack}\ [\sigma, e]\ \texttt{as}\ \exists \alpha.\, \tau \mid \texttt{let}\ [\alpha, x] = \texttt{unpack}\ e\ \texttt{in}\ e' \\ \text{Values} & v, w & ::= & \cdots \mid \texttt{pack}\ [\sigma, v]\ \texttt{as}\ \exists \alpha.\, \tau \end{array}$$

$$\frac{\Delta \vdash \sigma\ \text{type} \quad \Delta; \Gamma \vdash e : \tau[\sigma/\alpha]}{\Delta; \Gamma \vdash \texttt{pack}\ [\sigma, e]\ \texttt{as}\ \exists \alpha.\, \tau : \exists \alpha.\, \tau}$$

$$\frac{\Delta; \Gamma \vdash e : \exists \alpha.\, \tau \quad \Delta, \alpha; \Gamma, x : \tau \vdash e' : \tau' \quad \alpha \notin \mathrm{FV}(\tau')}{\Delta; \Gamma \vdash \texttt{let}\ [\alpha, x] = \texttt{unpack}\ e\ \texttt{in}\ e' : \tau'}$$

$$\frac{e \leadsto e'}{\texttt{pack}\ [\sigma, e]\ \texttt{as}\ \exists \alpha.\, \tau \leadsto \texttt{pack}\ [\sigma, e']\ \texttt{as}\ \exists \alpha.\, \tau}$$

$$\frac{e \leadsto e'}{\texttt{let}\ [\alpha, x] = \texttt{unpack}\ e\ \texttt{in}\ f \leadsto \texttt{let}\ [\alpha, x] = \texttt{unpack}\ e'\ \texttt{in}\ f}$$

$$\frac{}{\texttt{let}\ [\alpha, x] = \texttt{unpack}\ (\texttt{pack}\ [\sigma, v]\ \texttt{as}\ \exists \alpha.\, \tau)\ \texttt{in}\ e \leadsto e[\sigma/\alpha][v/x]}$$

2 A Simple Parametricity Result

We are working in plain call-by-value System F. Suppose that we are given a closed value f of type $\forall \alpha.\, (\alpha \to \alpha) \to (\alpha \to \alpha)$, a closed type τ, and two closed values v_s and v_z of type $\tau \to \tau$ and τ, respectively.

Your task is to define a set S that precisely characterizes the possible values that $f[\tau](v_s)(v_z)$ might compute to, but whose definition does NOT mention f. That is, for any value v, if it is possible that $f[\tau](v_s)(v_z) \leadsto^* v$, then $v \in S$, *and vice versa*. (If I left out "and vice versa", then you could just pick S to be the universal set.) Prove that your definition of S is correct.

3 Termination in the Presence of Constants of Arbitrary Type

Prove that call-by-value System F is still terminating even when extended with constants at every type (à la Girard). For the purpose of this problem, you can assume that the language has no base type **T** and no other constants beside 0 (see below). This illustrates that the existence of a value of the "false" type $\forall\alpha.\,\alpha$ does not break termination *per se*.

Here are the appropriate extensions:

$$\begin{array}{lrcl} \text{Terms} & e, f & ::= & \cdots \mid 0 \\ \text{Values} & v, w & ::= & \cdots \mid 0 \mid 0[\tau] \end{array}$$

$$\frac{}{\Delta;\Gamma \vdash 0 : \forall\alpha.\,\alpha}$$

$$\frac{}{0[\sigma \to \tau](v) \leadsto 0[\tau]} \qquad \frac{}{0[\forall\alpha.\,\tau][\sigma] \leadsto 0[\tau[\sigma/\alpha]]}$$

4 Using Girard's $\mathcal{J}$ to Implement Recursion

For this problem, we are working in System F with full reduction, extended with Girard's 0 and $\mathcal{J}$ operators. Recall the semantics of Girard's $\mathcal{J}$ operator:

$$\frac{}{\Delta;\Gamma \vdash \mathcal{J} : \forall\alpha.\forall\beta.\,\alpha \to \beta}$$

$$\frac{\sigma = \tau}{\mathcal{J}[\sigma][\tau](e) \leadsto e} \qquad \frac{\sigma \neq \tau \quad \sigma \text{ and } \tau \text{ are closed}}{\mathcal{J}[\sigma][\tau](e) \leadsto 0[\tau]}$$

Let us say that a closed term Y "encodes the fixed-point combinator at type τ" if: (1) Y has type $(\tau \to \tau) \to \tau$, and (2) for any closed term f of type $\tau \to \tau$, there exists a term e such that $Y(f) \leadsto^* e$ and $e \leadsto^* f(e)$.

Your task is to use Girard's $\mathcal{J}$ operator to define a closed term `fix` such that for all closed types τ, it is the case that `fix`$[\tau]$ encodes the fixed-point combinator at type τ.

Hint: $Y = \lambda f.\,(\lambda x.\,f(x\,x))(\lambda x.\,f(x\,x))$ is the fixed-point combinator in the classical untyped λ-calculus. At least in my solution to this problem, the untyped erasure of my `fix` is precisely the classical Y combinator.

5 Extra Credit: The $\mathcal{DJ}$ Is a Real Smooth Operator

Let us say that an operator *Op* of type σ "satisfies the Harper-Mitchell criterion" if: (1) there exist closed terms of type σ in pure System F, and (2) adding *Op* to System F causes strong normalization to fail. Furthermore, let us say that *Op* is a "real smooth operator" if: (1) it satisfies the Harper-Mitchell criterion, and (2) for all operators *Lame* that satisfy the Harper-Mitchell criterion, for all closed types τ, if *Lame* can be used to construct a term that encodes the fixed-point operator at type τ, then so can *Op*. In other words, *Op* is real smooth if it is the most powerful operator (in terms of encoding recursion) that one can define without violating the Harper-Mitchell criterion.

Prove that there exists a real smooth operator by defining one and proving that it is real smooth. (Mine is called $\mathcal{DJ}$, short for "Dreyer's $\mathcal{J}$".)