Apellido y Nombre:
email:

nota

1	2	3	4	5	6

Lenguajes y Compiladores **Parcial 3** **23/6/2009**

(1) Probar o refutar:
  (a) $[\![\langle 1, \textbf{letrec } f \equiv \lambda x.fx \textbf{ in } f\ x\rangle]\!]^{eager}_{\eta} = \bot$.
  (b) $[\![\langle 1, \textbf{letrec } f \equiv \lambda x.fx \textbf{ in } f\ x\rangle]\!]^{normal}_{\eta} = \bot$.
  (c) $[\![\textbf{sumcase } @\ 0\ e \textbf{ of } (f_0, f_1)]\!]^{eager}_{\eta} = [\![f_0\ e]\!]^{eager}_{\eta}$.
  (d) $[\![\textbf{sumcase } @\ 0\ e \textbf{ of } (f_0, f_1)]\!]^{normal}_{\eta} = [\![f_0\ e]\!]^{normal}_{\eta}$.
  (e) $[\![(\lambda\langle\rangle.e)\ \langle\rangle]\!]\eta\kappa = [\![e]\!]\eta\kappa$

(2) Probar que **callcc** $(\lambda k.\ 1 + \textbf{throw } k\ 1)$ es equivalente a la expresión 1.
¿Es **callcc** $(\lambda k.\ \textbf{True} + \textbf{throw } k\ 1)$ también equivalente a esa expresión?

(3) Considere la siguiente expresión:

$$\begin{array}{l} \textbf{let } f \equiv \lambda\langle x, g\rangle.\ \textbf{if } x = 1 \textbf{ then } g\langle 0, 1+g\rangle \textbf{ else } \langle 100 \div x, g\rangle \\ \textbf{in } f\langle 1, f\rangle \end{array}$$

(a) Elimine patrones y **let**.
(b) Sugiera (sin efectuar el cómputo) el valor de la semántica denotacional directa de esta expresión considerando por separado los casos eager y normal.

(4) Calcule la semántica operacional del programa:

$$\begin{array}{l} \textbf{letrec } f \equiv \lambda x. \\ \qquad \textbf{if } (\textbf{mkref } 0) =_{ref} (\textbf{mkref } 0) \textbf{ then true else } f(x-1) \\ \textbf{in } f \end{array}$$

(5) Demostrar el teorema de coincidencia para los casos 0, suma, abstracción, asignación del lenguaje Iswim.
Teorema de Coincidencia: Sean η y η' tales que $\forall w \in FV(e).\eta w = \eta' w$ entonces $[\![e]\!]\eta\kappa\sigma = [\![e]\!]\eta'\kappa\sigma$ para todo κ y σ.

(6) Para el lenguaje Iswim, sean v_0 y v_1 variables diferentes tales que $v_0 \notin FV(e_1)$ y $v_1 \notin FV(e_0)$. Demostrar o refutar la equivalencia del comando $v_0 := e_0; v_1 := e_1$ con el comando $v_1 := e_1; v_0 := e_0$.