

Lenguajes y Compiladores - TP 10 Continuación - Año 2012

Contenidos: *Un sistema de tipos simple.*

- (1) Para cada una de las siguientes expresiones determinar si la misma tiene un juicio de tipos válido. Si lo tiene encontrarlo, si no lo tiene explicar por qué.
 - (a) $(\lambda \langle x, y \rangle . x) \langle 3, \mathbf{true} \rangle$
 - (b) $\lambda p. \lambda x. \mathbf{if} \ p \ x \ \mathbf{then} \ x \ \mathbf{else} \ 0$
 - (c) $(\lambda x. x)(\lambda x. x)$
 - (d) $(\lambda f. \lambda x. f(f \ x))(\lambda x. x)$
 - (e) $\mathbf{let} \ i \equiv \lambda x. x \ \mathbf{in} \ i \ i$
- (2) Considere la siguiente expresión:

```
letrec taut  $\equiv \lambda n. \lambda f.$   
  if  $n = 0$  then  $f$  else  
    if  $taut(n - 1)(f \ \mathbf{true})$  then  $taut(n - 1)(f \ \mathbf{false})$   
    else  $\mathbf{false}$   
in taut
```

Puede esta expresión ser tipada usando el sistema de tipos dado en clase? Si la respuesta es afirmativa, dar el tipo, sino justificar por qué.
- (3) Dar la semántica intrínseca de **rec** en el lenguaje aplicativo tipado.
- (4) ¿Cómo modificaría la semántica intrínseca del lenguaje aplicativo tipado para que el lenguaje sea eager?
- (5) Probar que si $\pi \vdash e : \theta$ y $v \notin FV(e)$ entonces $\pi, v : \theta' \vdash e : \theta$.
- (6) Demostrar que el operador de punto fijo Y no tiene tipo.
- (7) Considere las reglas de tipado explícito. Dar una prueba directa utilizando inducción en la derivación de la siguiente proposición: si $\pi \vdash P : \theta_1$ y $\pi \vdash P : \theta_2$ entonces $\theta_1 = \theta_2$.