

Apellido y Nombre:

email:

nota
------

1	2	3	4	5	6
---	---	---	---	---	---

Lenguajes y Compiladores

Parcial 1

16/4/2009

- a) El teorema de sustitución para el lenguaje imperativo simple dice  
Si  $\delta$  es inyectiva sobre  $FV(c)$  y para todo  $w \in FV(c)$   $\sigma(\delta w) = \sigma'w$  entonces,  
1) o bien  $\llbracket c/\delta \rrbracket \sigma = \perp$  y  $\llbracket c \rrbracket \sigma' = \perp$ ,  
2) o bien  $\llbracket c/\delta \rrbracket \sigma \neq \perp$  y  $\llbracket c \rrbracket \sigma' \neq \perp$  y para todo  $w \in FV(c)$   $\llbracket c/\delta \rrbracket \sigma(\delta w) = \llbracket c \rrbracket \sigma'w$   
¿Cómo debería ser el enunciado para el lenguaje imperativo simple con fallas?  
b) Enunciar el Teorema de Renombre para el lenguaje imperativo simple.  
c) Se desea probar el Teorema enunciado en 1b por inducción estructural sobre  $c$  (el cuerpo de **newvar**  $v := e$  **in**  $c$ ). Desarrollar en detalle el caso  $c \equiv (w := e_1)$ . Enunciar todo resultado extra que utilice.

- Para la siguiente función  $F : (\mathbf{Z} \rightarrow \mathbf{Z}_\perp) \rightarrow (\mathbf{Z} \rightarrow \mathbf{Z}_\perp)$ , determinar el menor punto fijo utilizando el Teorema del punto fijo.

$$Ffn = \begin{cases} 0 & n = 0, \\ f(n - 2) + 1 & n > 0, n \text{ par} \\ f(-n) & c.c. \end{cases}$$

- Considere el programa

**while**  $x > 0$  **do if**  $x < 3$  **then**  $x := x - 1$  **else**  $!x$ ; **fail**

- a) Calcule  $F^k \perp$  para todo  $k \geq 0$ .  
b) Calcule la semántica denotacional.

- Para el lenguaje imperativo simple, demostrar la equivalencia

$c$ ; **if**  $b$  **then**  $c_0$  **else**  $c_1 \equiv$  **if**  $b$  **then**  $c$ ;  $c_0$  **else**  $c$ ;  $c_1$  .

donde  $c$  es el comando

$x := u$ ;  
 $y := w$ ;  
**newvar**  $u := u + 1$  **in**  
**while**  $u < y$  **do**  $(u := u + 1; y := y - 1)$

y  $b$  es la frase booleana  $u + w > 0$ .

- Demostrar o refutar

- a) **newvar**  $x := e$  **in**  $!x \equiv !e$ .  
b) **if**  $x > 0$  **then**  $!x$  **else**  $!(-x) \equiv$  (**if**  $x > 0$  **then skip** **else**  $x := -x$ );  $!x$

- Definir la semántica de **catchin**  $c_0$  **with**  $c_1$  para el lenguaje imperativo simple con fallas y output.