

Apellido y Nombre:

email:

nota
------

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

## Lenguajes y Compiladores

Primer Parcial 6/5/2016

- Determinar si es verdadero o falso. Justificar la respuesta.
  - Sean  $p, q$  predicados. Si  $\llbracket p \rrbracket = \llbracket q \rrbracket$ , entonces para toda sustitución  $\delta$  se tiene  $\llbracket p/\delta \rrbracket = \llbracket q/\delta \rrbracket$ .
  - Sea  $\Omega$  el dominio del lenguaje imperativo con fallas y output. Si  $\sigma$  es un estado, entonces existe una cadena interesante que tiene como supremo a  $\iota_{out}(1, \iota_{term}\sigma)$ .
  - Sea  $f, g \in \mathbf{Z} \rightarrow \mathbf{Z}_\perp$ . Entonces existe  $h \in \mathbf{Z} \rightarrow \mathbf{Z}_\perp$  tal que  $f \leq h$  y  $g \leq h$ .
  - En el lenguaje imperativo simple, si  $\llbracket c \rrbracket \sigma = \langle \mathbf{abort}, \sigma' \rangle$ , entonces  
$$\llbracket \mathbf{catchin} \ c \ \mathbf{with} \ c' \rrbracket \sigma = \llbracket c; c' \rrbracket \sigma.$$
- Considere el lenguaje aplicativo con fallas, output e input. Analice utilizando la semántica denotacional la equivalencia entre los siguientes comandos:
  - $\mathbf{newvar} \ v := e \ \mathbf{in} \ ?v; !v \equiv ?v; !v$
  - Si  $FA \ c \cap FA \ c' = \emptyset$  entonces  $c; c' \equiv c'; c$
- Considere el lenguaje imperativo simple.
  - Dé la semántica denotacional de  $\mathbf{while} \ b \ \mathbf{do} \ c$ .
  - Pruebe que la función  $F$  que define la semántica de  $\mathbf{while} \ b \ \mathbf{do} \ c$  es continua.
  - De ejemplo de un comando  $c$  de la forma  $\mathbf{while} \ b \ \mathbf{do} \ c$  tal que  $\llbracket c \rrbracket = F^3 \perp_{\Sigma \rightarrow \Sigma_\perp}$  pero  $\llbracket c \rrbracket \neq F^2 \perp_{\Sigma \rightarrow \Sigma_\perp}$ .
- Considere el lenguaje imperativo simple, y sea  $c = c_0; c_1$ .
  - De las reglas de la semántica smallstep  $\rightarrow$  para esta frase abstracta.
  - Demuestre que si  $\llbracket c \rrbracket \sigma = \sigma'$ , entonces  $\langle c, \sigma \rangle \rightarrow^* \sigma'$ , para esta frase abstracta. Enuncie (sin probar) todo resultado que utilice.

5. Considere la función  $F : (\mathbf{Z} \rightarrow \mathbf{Z}_\perp) \rightarrow (\mathbf{Z} \rightarrow \mathbf{Z}_\perp)$  dada por:

$$Ffn = \begin{cases} n & n = 0, 1, 2 \\ f(n-3) & n > 1 \\ f(-n) & n < 0 \end{cases}$$

- a) ¿Cuánto vale  $F^5 \perp_{\mathbf{Z} \rightarrow \mathbf{Z}_\perp} (-10)$ ?  
 b) ¿Cuánto vale el menor punto fijo de  $F$  en  $-10$ ? Justifique su respuesta.  
 c) Justifique la siguiente afirmación:  $F^2 \perp_{\mathbf{Z} \rightarrow \mathbf{Z}_\perp} \leq F^3 \perp_{\mathbf{Z} \rightarrow \mathbf{Z}_\perp}$ .  
 d) Pruebe que  $F$  es continua.

6. Considere el lenguaje imperativo simple.

- a) De las reglas de la semántica smallstep  $\rightarrow$  para al comando **newvar**  $v := e$  **in**  $c_0$ .  
 b) Demuestre que si  $\llbracket c \rrbracket \sigma = \sigma'$ , entonces  $\langle c, \sigma \rangle \rightarrow^* \sigma'$ , para el caso  
 $c = \mathbf{newvar} \ v := e \ \mathbf{in} \ c_0$ .

Enuncie (sin probar) todo resultado que utilice.

7. a) Complete las siguientes igualdades, expresando de la forma más sencilla posible el resultado, sin efectuar ningún cálculo. Considere el lenguaje que corresponde en cada caso.

- 1)  $\llbracket \forall x. \exists y. y + y = y \rrbracket \sigma =$   
 2)  $\llbracket ?x; \mathbf{while} \ \mathbf{true} \ \mathbf{do} \ \mathbf{skip}; !x \rrbracket \sigma =$   
 3)  $\llbracket x := 1; \mathbf{newvar} \ x := 0 \ \mathbf{in} \ (!x; \mathbf{fail}; !x) \rrbracket \sigma =$

- b) Calcule la semántica denotacional del programa del item a) 3).

8. Considere el dominio  $\Omega$  del lenguaje con fallas, input y output.

- a) ¿Qué relaciones de orden encuentra entre los siguientes elementos?

$$\iota_{in}(\perp_{\mathbf{Z} \rightarrow \Omega}), \ \iota_{out}(n, \iota_{term} \sigma), \ \iota_{in}(\lambda n \in \mathbf{Z}. \iota_{term} \sigma), \ \iota_{out}(n, \perp), \\ \iota_{out}(n, \iota_{out}(n, \iota_{term} \sigma)), \ \iota_{in}(\lambda n \in \mathbf{Z}. \iota_{abort} \sigma),$$

- b) Dé un ejemplo de una cadena interesante cuyo primer elemento sea  $\iota_{in}(\perp_{\mathbf{Z} \rightarrow \Omega})$ .  
 c) ¿Puede encontrar un programa que tenga como semántica al supremo de la cadena?  
 Si la respuesta es sí, muétrelo. (No calcule nada!)

9. Determinar si son equivalentes. Si lo son probarlo utilizando semántica denotacional, si no lo son, dar un contraejemplo.

**newvar**  $v := e$  **in** **catchin**  $c_0$  **with**  $c_1$   
**catchin** **newvar**  $v := e$  **in**  $c_0$  **with**  $c_1$