

**Apellido y Nombre:**  
**email (@mi.unc.edu.ar):**  
**Nota:**

### Lenguajes y Compiladores

### Examen Final 01-10-2024

1. Considerá la siguiente ecuación con  $f : \mathbb{Z} \rightarrow \mathbb{Z}$ :

$$f x = \begin{cases} x & \text{si } 10 \leq x \\ f(x+1) & \text{si } 0 < x < 9 \\ f(x-1) & \text{si } x \leq 0 \end{cases}$$

- a) Proponé dos soluciones distintas en  $\mathbb{Z} \rightarrow \mathbb{Z}$ .
- b) ¿Tiene una menor solución esa ecuación en  $\mathbb{Z} \rightarrow \mathbb{Z}$ ?
- c) ¿Si existe esa menor solución, podemos usar el teorema de menor punto fijo en  $\mathbb{Z} \rightarrow \mathbb{Z}$  para encontrarla?

2. Considerá el lenguaje imperativo con input/output y fallas.

$$F: (\Sigma \rightarrow \Omega) \rightarrow (\Sigma \rightarrow \Omega)$$

$$F f \sigma = \begin{cases} \iota_{in}(z \mapsto \iota_{out}(z, f[\sigma | x : z])) & \text{si } \sigma x = 0 \\ \iota_{in}(z \mapsto \iota_{out}(z, f[\sigma | x : \sigma x - 1])) & \text{si } \sigma x > 0 \\ \iota_{abort} \sigma & \text{si } \sigma x < 0 \end{cases}$$

- a) Proponé un programa de la forma **while**  $b$  **do**  $c$  cuyo funcional sea  $F$ .
  - b) Sea  $\sigma_k$  un estado tal que  $\sigma_k x = k$ . ¿Vale la igualdad  $F^i \perp \sigma_2 = F^i \perp \sigma_3$  para todo  $i$ ?
  - c) ¿Hay algún estado  $\sigma$  tal que  $F^2 \perp \sigma = \iota_{in}(z \mapsto \iota_{out}(z, \iota_{abort}[\sigma | x : z]))$ ?
3. Considerá la expresión  $e = \lambda y. y ((\lambda x. x) y)$ . Respondé si es verdadero, falso, o *impreciso* cada ítem justificando tu respuesta.

- a) En el calculo lambda puro,  $e(ue)$  tiene forma normal. Aquí  $u$  es una variable cualquiera.
  - b) En el calculo con evaluación normal, para cualquier expresión  $e'$  existe  $z$  (forma canónica) tal que  $ee' \Rightarrow_N z$ .
  - c) En el calculo con evaluación eager,  $e \Rightarrow_E (\lambda y. y y)$  y por lo tanto  $ee$  no tiene forma canónica.
4. Considerá los lenguajes aplicativos eager y normal.

- a) Considerá la siguiente ecuación recursiva  $t = \iota_{tuple}\langle 0, t \rangle$ . Proponé una expresión en el lenguaje aplicativo normal cuya semántica sea una solución para  $t$ .
  - b) Considerá la expresión **letrec**  $f \equiv \lambda u. \langle 0, \lambda v. f 0 \rangle$  **in**  $f$ . Calculá la semántica eager de dicha expresión; para ello, primero expresá el funcional asociado de la manera más sencilla posible.
5. Proponé una expresión  $e$  que tenga como única variable libre a  $n$  tal que  $\llbracket e \rrbracket [\eta | n : \iota_{int} k] \rrbracket$  sea un estado con  $\max(k, 0)$  referencias todas accesibles a partir del valor devuelto. Por ejemplo, con  $k = 1$  la semántica podría ser  $\iota_{norm} \langle [r_0 : \iota_{tuple} \langle \rangle], \iota_{ref} r_0 \rangle$ .

#### 6. Ejercicio para libres:

- a) Proponé una expresión  $fact_N$  cuya semántica sea la función factorial en lenguaje normal. Es decir  $\llbracket fact_N \rrbracket \eta = \iota_{fun} f$  tal que  $f(\iota_{nat} k) = \iota_{nat} k!$  si  $k \geq 0$ .
- b) Proponé una expresión  $fact_E$  cuya semántica sea la función factorial en el lenguaje eager.
- c) Explicá con tus palabras la diferencia entre los operadores de recursión de esos dos lenguajes.