

Lenguajes y Compiladores - Guía del 3/06/2016

Objetivos: Comprender las extensiones del lenguaje aplicativo eager con referencias y asignación. Poder relacionar las nociones operacionales con las denotacionales. Comprender las definiciones de los comandos usuales de un lenguaje imperativo en el lenguaje aplicativo extendido con referencias y asignación.

1. Evaluar (semántica operacional) las siguientes expresiones en el lenguaje aplicativo eager con referencias y asignación, en un entorno cualquiera η y un estado σ :

$r := 1; 1 + \mathbf{val} r \quad (r \in \langle \mathit{refcnf} \rangle)$

$\mathbf{newvar} x := 1 \mathbf{in} ((1 + \mathbf{val} x); x := 2 + \mathbf{val} x)$

$\mathbf{while} \mathbf{True} \mathbf{do} \mathbf{skip}$

2. Evaluar la expresión $\mathbf{newvar} x := \mathbf{mkref} 0 \mathbf{in} e$ en el estado \square , para los siguientes e :

a. $(\mathbf{val} x) := 1$ b. $x := 1$ c. $(\mathbf{val} x) := \mathbf{val}(\mathbf{val} x)$

3. Para el lenguaje aplicativo eager con referencias y asignación, calcular la semántica de denotacional de $x := 1 + \mathbf{val} x$ en:

a) η, σ tales que $\eta x = \iota_{\mathit{ref}} r, \sigma r = \iota_{\mathit{int}} 1$.

b) η, σ tales que $\eta x = \iota_{\mathit{ref}} r, \sigma r = \iota_{\mathit{bool}} \mathbf{True}$.

c) η, σ tales que $\eta x = \iota_{\mathit{int}} 0$.

4. Considere la expresión

$\mathbf{let} y \equiv \mathbf{ref} 0 \mathbf{in} (\lambda x.(y := 1; x))(\mathbf{val} y)$.

Dar el resultado de evaluar la expresión, y también el resultado de evaluar la expresión contrayendo previamente la (única) redex β .

5. Calcule la semántica denotacional de los programas del ejercicio 3.