

Fondamenti di Linguaggi di Programmazione 19/06/2006

Esercizio 1 (8 punti)

Si consideri un nuovo comando IMP

let $x = a$ **in** c

la cui semantica informale è quella di eseguire il comando c con la variabile x (che precedentemente conteneva m) inizializzata al valore n dell'espressione aritmetica a , e di riassegnare m ad x dopo aver terminato l'esecuzione di c . Si dia

- la semantica operativa e la semantica denotazionale del nuovo comando;

Soluzione. Semantica operativa:

$$\frac{\langle x, \sigma \rangle \rightarrow m \quad \langle a, \sigma \rangle \rightarrow n \quad \langle c, \sigma[n/x] \rangle \rightarrow \sigma'}{\langle \mathbf{let} x = a \mathbf{in} c, \sigma \rangle \rightarrow \sigma'[m/x]}$$

Semantica denotazionale:

$$\begin{aligned} \mathcal{C}[\mathbf{let} x = a \mathbf{in} c] &= \\ &= \{(\sigma, \sigma') \mid \mathcal{A}[x] = (\sigma, m) \ \& \ \mathcal{A}[a] = (\sigma, n) \ \& \ \mathcal{C}[c] = (\sigma[n/x], \sigma'') \ \& \ \sigma' = \sigma''[m/x]\} \end{aligned}$$

- si estendano le dimostrazioni di equivalenza tra le semantiche per tenere conto del nuovo comando.

Soluzione. Tenendo conto che c è un comando di **IMP** e che la valutazione delle espressioni aritmetiche e dei comandi di **imp** è deterministica, bisogna provare che

$$\forall \sigma, \sigma' \in \Sigma, \quad \langle \mathbf{let} x = a \mathbf{in} c, \sigma \rangle \rightarrow \sigma' \iff \mathcal{C}[\mathbf{let} x = a \mathbf{in} c]\sigma = \sigma'$$

Esercizio 2 (6 punti)

Siano (D, \sqsubseteq_D) e (E, \sqsubseteq_E) due cpo. Consideriamo l'insieme intersezione $D \cap E$ e la seguente relazione \sqsubseteq su $D \cap E$:

$$x \sqsubseteq y \quad \mathbf{iff} \quad x \sqsubseteq_D y \ \mathbf{e} \ x \sqsubseteq_E y$$

Supponendo che $D \cap E \neq \emptyset$, dire se $(D \cap E, \sqsubseteq)$ è un cpo, motivando formalmente le risposte.

Esercizio 3 (8 punti)

Verificare quali dei seguenti termini é tipabile. Per ogni termine tipabile si mostri il tipo e la forma canonica (eager e lazy).

1. **rec last.** $\lambda x.$ **if** $\text{snd}(x)$ **then** $\text{fst}(x)$ **else** $(\text{last } \text{snd}(x))$
2. **if** $\text{fst}(x) - \text{snd}(x)$ **then** $\text{fst}(x)$ **else** $\text{snd}(x)$
3. **if** $\text{fst}(x) - \text{snd}(x)$ **then** x **else** $x + x$

Soluzione: Il primo e il terzo non sono tipabili.

Esercizio 4 (8 punti)

Si dimostri in modo formale che i due comandi:

while $x \neq 0$ **do** **if** $x < 0$ **then** $x := -x$ **else** $x := 0$ **e** $x := 0$

hanno la stessa semantica (operazionale e denotazionale).