

Linguaggi di Programmazione

Esame-1 Giugno 2022

Domanda

Qual e' la formula per esprimere l'equivalenza nella semantica denotazionale?

Esercizio 1

Estendere IMP con il comando

$$\mathbf{try } c_1 = c_2 \mathbf{ else } c_3$$

che restituisce la memoria ottenuta calcolando c_1 se coincide con quella ottenuta calcolando c_2 , la memoria ottenuta calcolando c_3 se differiscono e diverge altrimenti.

1. Definire la semantica operativa del nuovo comando.
2. Definire la semantica denotazionale del nuovo comando.
3. Estendere la prova di corrispondenza tra la semantica operativa e quella denotazionale.

Esercizio 2

1. Definire in FSharp il tipo dato "albero ternario di interi", cioè una struttura dati ad albero che in ogni nodo contiene un intero ed eventualmente esattamente tre figli (a meno che il nodo non sia una foglia). Consiglio: trarre ispirazione dalla definizione delle liste in FSharp, che possono essere vuote oppure contenere un elemento e una lista.
2. Dare un piccolo esempio di albero definito in questa maniera, e disegnare la sua rappresentazione grafica (nodi e archi).
3. Definire una funzione ricorsiva che calcola il prodotto dei valori contenuti in un albero, valutando 1 le foglie.

Esercizio 3

Dimostrare che i termini HOFL:

$$t_1 \stackrel{\text{def}}{=} \mathbf{rec } f.\lambda x. ((\lambda y. 1) (f x)) \quad t_2 \stackrel{\text{def}}{=} \lambda x. 1$$

hanno lo stesso tipo e la stessa semantica denotazionale.

Esercizio 4

Estendere l'interprete definito a lezione, traendo spunto dai domini sintattici e semantici nel prontuario.

1. Definire il comando `CRepeatUntilZero(ide, pseq)` che prende come argomenti un identificatore e una sequenza di comandi. La semantica intesa è: si esegue la sequenza `pseq`, e poi si verifica se la variabile denotata dall'identificatore `ide` ha il valore 0. Se l'identificatore non denota una variabile la semantica non è definita (pertanto questa è una condizione di errore). Se non è zero, il ciclo viene ripetuto; se è zero, il ciclo termina. Le modifiche allo stato vanno preservate nello stato di uscita, ma non le modifiche all'ambiente (usiamo il binding statico). Opzionale: fare in modo che le locazioni eventualmente allocate nella sequenza siano cancellate quando non sono più utili. Nota bene: la sequenza `pseq` è sempre eseguito almeno una volta.

Si fornisca il caso aggiuntivo della sintassi astratta (e a quale tipo va aggiunto) e il caso aggiuntivo della funzione di valutazione semantica dei comandi, che ha il seguente tipo e struttura:

```
let rec csem: com -> env -> store -> (env * store) =
  fun c ev st ->
    match c with
    | Cassign (i, e) ->
      ...

and pssem: pseq -> env -> store -> (env * store) =
  fun s ev st ->
    ...

and ...
```

2. Dare un esempio di uso, in cui il programma termina, e si vede almeno una ripetizione del ciclo. Non è necessario scrivere l'intero programma, solo il comando `CRepeatUntilZero(ide, pseq)` con `ide` e `pseq` opportunamente istanziati, assumendo che `ide` denoti una variabile.

Esercizio 5

Consideriamo gli agenti CCS:

$$p \stackrel{\text{def}}{=} (\text{rec } x. a.x) | \text{rec } x. b.x \quad q \stackrel{\text{def}}{=} \text{rec } x. a.a.x + a.b.x + b.a.x + b.b.x$$

Dimostrare che p e q sono fortemente bisimili o mostrare una formula HM-logica F che possa essere usata per distinguerli.