



Fondamenti dei linguaggi di programmazione

Aniello Murano
Università degli Studi di Napoli
"Federico II"

Murano Aniello
Fond. LP - Decima Lezione

1



**Equivalenza delle semantiche
operazionale e denotazionale
dei comandi di IMP**

Murano Aniello
Fond. LP - Decima Lezione

2

Semantica denotazionale

- Al linguaggio IMP sono associate le seguenti funzioni:
 - $\mathcal{A}: Aexp \rightarrow (\Sigma \rightarrow \mathcal{N})$;
 - $\mathcal{B}: Bexp \rightarrow (\Sigma \rightarrow \{\text{true}, \text{false}\})$,
 - $\mathcal{C}: Com \rightarrow (\Sigma \rightarrow \Sigma)$
- Per esempio, se la denotazione di X è data da $(\sigma, \sigma(X))$, la valutazione di X in uno stato σ è data da $\mathcal{A}[[X]](\sigma) = \sigma(X)$
- Per i comandi, questo ragionamento può dare luogo a difficoltà.
- Per esempio, sia $w \equiv \text{while } b \text{ do } c$. Abbiamo visto che $\mathcal{C}[[w]] = \{(\sigma, \sigma') \mid \mathcal{B}[[b]]\sigma = \text{true} \ \& \ (\sigma, \sigma') \in \mathcal{C}[[w]] \circ \mathcal{C}[[c]]\} \cup \{(\sigma, \sigma) \mid \mathcal{B}[[b]]\sigma = \text{false}\}$ non da la denotazione di w , perché $\mathcal{C}[[w]]$ è una funzione ricorsiva.
- In pratica, la denotazione di w è data dalla sua denotazione al prossimo ciclo concatenata alla denotazione del comando c .
- Questo ragionamento suggerisce una funzione continua $f: (\Sigma \rightarrow \Sigma) \rightarrow (\Sigma \rightarrow \Sigma)$ il cui fixpoint (corrispondente all'eventuale termine del ciclo while) fornisce la denotazione di w .



Murano Aniello
Fond. LP - Decima Lezione

3

Fixpoint per $\mathcal{C}[[w]]$

Sia $\mathcal{C}[[w]] =$
 $\{(\sigma, \sigma') \mid \mathcal{B}[[b]]\sigma = \text{true} \ \& \ (\sigma, \sigma') \in \mathcal{C}[[w]] \circ \mathcal{C}[[c]]\} \cup$
 $\{(\sigma, \sigma) \mid \mathcal{B}[[b]]\sigma = \text{false}\} =$
 $\{(\sigma, \sigma') \mid \mathcal{B}[[b]]\sigma = \text{true} \ \& \ (\sigma, \sigma'') \in \mathcal{C}[[c]] \ \& \ (\sigma'', \sigma') \in \mathcal{C}[[w]]\} \cup$
 $\{(\sigma, \sigma) \mid \mathcal{B}[[b]]\sigma = \text{false}\}$

f è dunque definita sulle seguenti regole

$$\frac{\perp}{\sigma, \sigma} \quad \text{se } \mathcal{B}[[b]]\sigma = \text{false}$$

$$\frac{\sigma'', \sigma'}{\sigma, \sigma'} \quad \text{se } \mathcal{B}[[b]]\sigma = \text{true} \ \& \ (\sigma, \sigma'') \in \mathcal{C}[[c]]$$



Murano Aniello
Fond. LP - Decima Lezione

4

Equivalenza delle semantiche

- In questa lezione mostriamo l'equivalenza della semantica operativa con quella denotazionale per il linguaggio IMP.
- Per provare l'equivalenza delle due semantiche occorre dimostrare che

- Per ogni $a \in \text{Aexp}$, $(\sigma, n) \in \mathcal{A}[[a]] \Leftrightarrow \langle a, \sigma \rangle \rightarrow n$ (già provato)
- Per ogni $b \in \text{Bexp}$, $(\sigma, t) \in \mathcal{B}[[b]] \Leftrightarrow \langle b, \sigma \rangle \rightarrow t$ (già provato)
- Per ogni $c \in \text{Com}$, $(\sigma, \sigma') \in \mathcal{C}[[c]] \Leftrightarrow \langle c, \sigma \rangle \rightarrow \sigma'$ (da provare)



Murano Aniello
Fond. LP - Decima Lezione

5

Equivalenza per Com (1)

- Dimostriamo che per ogni $c \in \text{Com}$ vale la proprietà $P(c)$
 $(\sigma, \sigma') \in \mathcal{C}[[c]] \Leftrightarrow \langle c, \sigma \rangle \rightarrow \sigma'$

- Iniziamo con il verso " \Leftarrow ".
- Per questo verso usiamo una induzione sulle derivazioni.

Com ::= skip | X:=a | c₀;c₁ | if b then c₀ else c₁ | while b do c

- Sia $c \equiv \text{skip}$. Per gli assiomi, $\langle \text{skip}, \sigma \rangle \rightarrow \sigma$ e $(\sigma, \sigma) \in \mathcal{C}[[\text{skip}]]$
- Sia $c \equiv X:=a$. Se $\langle X:=a, \sigma \rangle \rightarrow \sigma'$, allora esiste una derivazione

$$\frac{\dots}{\frac{\langle a, \sigma \rangle \rightarrow n}{\langle X:=a, \sigma \rangle \rightarrow \sigma'}}$$

con $\sigma' = \sigma[n/X]$.

Per l'equivalenza sulle espressioni di Aexp, segue $(\sigma, n) \in \mathcal{A}[[a]]$.

Dato che $\mathcal{C}[[X:=a]] = \{(\sigma, \sigma[n/X]) \mid \mathcal{A}[[a_0]]\sigma = n\}$, segue $(\sigma, \sigma') \in \mathcal{C}[[c]]$



Murano Aniello
Fond. LP - Decima Lezione

6

Equivalenza per Com di ":" per " \Leftarrow "

- Sia $c = c_0; c_1$

Se $\langle c_0; c_1, \sigma \rangle \rightarrow \sigma'$ allora esiste la seguente derivazione

$$\frac{\langle c_0, \sigma \rangle \rightarrow \sigma'' \quad \langle c_1, \sigma'' \rangle \rightarrow \sigma'}{\langle c_0; c_1, \sigma \rangle \rightarrow \sigma'}$$

Per ipotesi induttiva, abbiamo che $\mathcal{C}[\![c_0]\!] \sigma = \sigma''$ e $\mathcal{C}[\![c_1]\!] \sigma'' = \sigma'$.

Dunque:

$$\mathcal{C}[\![c_0; c_1]\!] \sigma = \mathcal{C}[\![c_1]\!] (\mathcal{C}[\![c_0]\!] \sigma) = \mathcal{C}[\![c_1]\!] \sigma'' = \sigma', \text{ come richiesto}$$

- Esercizio: Provare che la direzione " \Leftarrow " vale per il comando "if"



Murano Aniello
Fond. LP - Decima Lezione

7

Equivalenza per Com: while per " \Leftarrow "

- Sia $w \equiv \text{while } b \text{ do } c$. Proviamo che $(\sigma, \sigma') \in \mathcal{C}[\![w]\!] \Leftarrow \langle w, \sigma \rangle \rightarrow \sigma'$.
- Se $\langle w, \sigma \rangle \rightarrow \sigma'$, allora esistono due possibili derivazioni dipendenti dalla valutazione "true" o "false" di b . Nel primo caso di false:

$$\frac{\langle b, \sigma \rangle \rightarrow \text{false}}{\langle w, \sigma \rangle \rightarrow \sigma}$$

1. Per l'equivalenza provata su B_{exp} , $(\sigma, \text{false}) \in \mathcal{B}[\![b]\!]$.

- Ricordiamo che la denotazione di w (nel caso $(\sigma, \text{false}) \in \mathcal{B}[\![b]\!]$) è

$$\mathcal{C}[\![w]\!] = \{(\sigma, \sigma) \mid (\sigma, \sigma) \in \mathcal{B}[\![b]\!]\}$$

dunque il risultato è vero prendendo $\sigma' = \sigma$



Murano Aniello
Fond. LP - Decima Lezione

8

Equivalenza per Com: while per " \Leftarrow " (2)

- Per $w \equiv \text{while } b \text{ do } c$, stiamo provando che $(\sigma, \sigma') \in \mathcal{C}[[w]] \Leftarrow \langle w, \sigma \rangle \rightarrow \sigma'$.
- Se b è valutata "true", allora:

$$\frac{\langle b, \sigma \rangle \rightarrow \text{true} \quad d_1 ::= \langle c, \sigma \rangle \rightarrow \sigma'' \quad d_2 ::= \langle w, \sigma'' \rangle \rightarrow \sigma'}{\langle w, \sigma \rangle \rightarrow \sigma'}$$

- Per l'equivalenza provata su B_{exp} , $(\sigma, \text{true}) \in \mathcal{B}[[b]]$.
- Per ipotesi induttiva su d_1 , da $\langle c, \sigma \rangle \rightarrow \sigma''$ segue che $(\sigma, \sigma'') \in \mathcal{C}[[c]]$
- Per ipotesi induttiva su d_2 , da $\langle w, \sigma'' \rangle \rightarrow \sigma'$ segue che $(\sigma'', \sigma') \in \mathcal{C}[[w]]$

- Ricordiamo che la denotazione di w (nel caso $(\sigma, \text{true}) \in \mathcal{B}[[b]]$) è $\mathcal{C}[[w]] = \{(\sigma, \sigma_1) \mid (\sigma, \sigma_1) \in \mathcal{C}[[w]] \circ \mathcal{C}[[c]]\}$ il cui risultato è dato dal fixpoint dell'operatore di punto fisso f ...
- Siccome $\mathcal{C}[[w]]\sigma = (\mathcal{C}[[w]] \circ \mathcal{C}[[c]])\sigma = (\mathcal{C}[[c;w]])\sigma = \mathcal{C}[[w]](\mathcal{C}[[c]]\sigma)$, segue $\mathcal{C}[[w]]\sigma = \text{fix } \mathcal{C}[[w]]\sigma = \text{fix } \mathcal{C}[[w]](\mathcal{C}[[c]]\sigma) = \text{fix } \mathcal{C}[[w]]\sigma'' = \mathcal{C}[[w]]\sigma'' = \sigma'$ Dunque $(\sigma, \sigma') \in \mathcal{C}[[w]]$

- La prova è ancora più semplice se consideriamo le regole di punto fisso..



Equiv. su while per " \Leftarrow " (passo induttivo)

- La semantica denotazionale del comando $w \equiv \text{while } b \text{ do } c$ è data dal punto fisso dell'operatore f definito dalle seguenti regole:

$$\frac{}{\sigma, \sigma} \text{ se } \mathcal{B}[[b]]\sigma = \text{false} \quad \frac{\sigma'', \sigma'}{\sigma, \sigma'} \text{ se } \mathcal{B}[[b]]\sigma = \text{true} \ \& \ (\sigma, \sigma'') \in \mathcal{C}[[c]]$$

$$\text{dove } \mathcal{C}[[w]] = \text{fix } (f) = \bigsqcup_{n \in \mathbb{N}} f^n(\perp)$$

- Dobbiamo provare che $\langle w, \sigma \rangle \rightarrow \sigma' \Rightarrow (\sigma, \sigma') \in \mathcal{C}[[w]]$, con $\langle b, \sigma \rangle \rightarrow \text{True}$,

$$\frac{\langle b, \sigma \rangle \rightarrow \text{true} \quad \langle c, \sigma \rangle \rightarrow \sigma'' \quad \langle w, \sigma'' \rangle \rightarrow \sigma'}{\langle w, \sigma \rangle \rightarrow \sigma'}$$

- $(\sigma, \text{true}) \in \mathcal{B}[[b]]$ (per l'equiv. su B_{exp}) e $(\sigma, \sigma'') \in \mathcal{C}[[c]]$ (per induz. esterna)
- Per ipotesi indutt. su d_2 , $(\sigma'', \sigma') \in \mathcal{C}[[w]]$. Dato $\mathcal{C}[[w]] = \bigcup_{n \geq 0} f^n(\perp)$, segue che esiste un indice i tale che $(\sigma'', \sigma') \in f^i(\perp)$. Ma allora, posso applicare la regola iterativa dell'operatore di punto fisso ed ho che $(\sigma'', \sigma') \in f(f^i(\perp)) = f^{i+1}(\perp) \subseteq \bigcup_{i \geq 0} f^i(\perp) = \mathcal{C}[[w]]$



Equivalenza per Com "⇒"

- Dobbiamo dimostrare che

$$(\sigma, \sigma') \in \mathcal{C}[[c]] \Rightarrow \langle c, \sigma \rangle \rightarrow \sigma'$$

- L'equivalenza sui comandi

Skip, X:=a, $c_0;c_1$, if b then c_0 else c_1

è lasciata per esercizio.

- Proviamo il verso "⇒" solo per il comando while che è anche il più complesso



Equivalenza per Com: while per "⇒"

- Sia dunque $w \equiv \text{while } b \text{ do } c$ e vogliamo provare che

$$(\sigma, \sigma') \in \mathcal{C}[[w]] \Rightarrow \langle w, \sigma \rangle \rightarrow \sigma'$$

supponendo che l'equivalenza delle semantiche sia vera per c

- Si ricordi che $\mathcal{C}[[w]] = \{(\sigma, \sigma') \mid \mathcal{B}[[b]]\sigma = \text{true} \ \& \ (\sigma, \sigma') \in \mathcal{C}[[w]] \circ \mathcal{C}[[c]]\} \cup \{(\sigma, \sigma) \mid \mathcal{B}[[b]]\sigma = \text{false}\}$, la cui soluzione è data dal fix point
- A questo scopo, nella lezione precedente, abbiamo introdotto
 - > $\mathcal{C}_0[[w]] \equiv \perp$ come una funzione indefinita
 - > e $\forall n \geq 1, \mathcal{C}_n[[w]]$ la valutazione di while con al più n valutazioni di b.
- Inoltre abbiamo introdotto una funzioni f su $\mathcal{C}[[w]]$ tale che
 - > $f^0(\perp) = \perp = \mathcal{C}_0[[w]]$, $f^1(\perp) = f(f^0(\perp)) = \mathcal{C}_1[[w]]$,
 - > $\forall n > 1, f^n(\perp) = f(f^{n-1}(\perp)) = \mathcal{C}_n[[w]]$
 - > $\mathcal{C}[[w]] = \text{fix}(f) = \left(\bigsqcup_{n \in \omega} f^n(\perp) \right)$
- In pratica, $\mathcal{C}_n[[w]] = \{(\sigma, \sigma') \mid \mathcal{B}[[b]]\sigma = \text{true} \ \& \ (\sigma, \sigma') \in \mathcal{C}_{n-1}[[w]] \circ \mathcal{C}[[c]]\} \cup \{(\sigma, \sigma) \mid \mathcal{B}[[b]]\sigma = \text{false}\}$
- Se si dimostra che $\forall \sigma, \sigma' \in \Sigma. (\sigma, \sigma') \in \mathcal{C}_n[[w]] \Rightarrow \langle w, \sigma \rangle \rightarrow \sigma'$ allora per il fixpoint abbiamo che $(\sigma, \sigma') \in \mathcal{C}[[w]] \Rightarrow \langle w, \sigma \rangle \rightarrow \sigma'$.



Intuizione sulla formulazione della prova

- Intuitivamente, l'asserto che vogliamo provare è
 - Ogni denotazione (σ, σ') ottenuta con n iterazioni dell'operatore di punto fisso corrisponde ad un albero di derivazione del while di altezza n .
1. Base: Proviamo che questo è vero per il caso base: nessuna valutazione/denotazione del while,
 2. Passo induttivo: supponendo che l'asserto sia vero per un albero di derivazione alto n e per n cicli di valutazioni nella denotazione del while (con $n > 0$, quindi anche per 0 iterazioni), si prova che l'asserto è vero per $n+1$ (cioè aggiungendo un livello nell'albero di valutazione del while nella semantica operativa / aggiungendo una iterazione della valutazione del while nella semantica denotazione)
- La prova segue dal punto fisso (ogni possibile coppia denotazionale (σ, σ') corrisponde ad una valutazione da σ a σ')
 - Nel passaggio induttivo, chiaramente la valutazione può o meno portare ad una iterazione.



Murano Aniello
Fond. LP - Decima Lezione

13

Equivalenza per Com: while per "⇒" (2)

- Dimostriamo per induzione matematica che
- $$\forall \sigma, \sigma' \in \Sigma. (\sigma, \sigma') \in C_n \llbracket w \rrbracket \Rightarrow \langle w, \sigma \rangle \rightarrow \sigma'$$
- Caso base: $n = 0$. Vacuamente vera (si ricordi che $F \Rightarrow (T/F)$ è sempre vera). Infatti, (σ, σ') non appartiene a $C_0 \llbracket w \rrbracket = \perp$. Si noti come anche per $n=1$ la prova è semplice (basta considerare $\sigma = \sigma'$).
 - Supposto vero per un arbitrario numero naturale n , dimostriamo che $(\sigma, \sigma') \in C_{n+1} \llbracket w \rrbracket \Rightarrow \langle w, \sigma \rangle \rightarrow \sigma'$ per qualsiasi coppia di stati
 - Si assuma $(\sigma, \sigma') \in C_{n+1} \llbracket w \rrbracket$. Allora
 1. $B \llbracket b \rrbracket \sigma = \text{true}$ e $(\sigma, \sigma') \in C_n \llbracket w \rrbracket \circ C \llbracket c \rrbracket$ oppure
 2. $B \llbracket b \rrbracket \sigma = \text{false}$ e $\sigma = \sigma'$.
 - Per il caso 2, il risultato segue dalla semantica operativa di w
 - Per il caso 1, per l'equivalenza su Bexp segue $\langle b, \sigma \rangle \rightarrow \text{true}$, inoltre $(\sigma, \sigma'') \in C \llbracket c \rrbracket$ implica $\langle c, \sigma \rangle \rightarrow \sigma''$ e per ipotesi induttiva, $(\sigma'', \sigma') \in C_n \llbracket w \rrbracket$ implica $\langle w, \sigma'' \rangle \rightarrow \sigma'$.
 - Per la regola di derivazione del while otteniamo $\langle w, \sigma \rangle \rightarrow \sigma'$



Murano Aniello
Fond. LP - Decima Lezione

14

Esercizio 1

- Dare la semantica operativa e denotazionale del seguente comando e dimostrare la loro equivalenza.
- Si consideri il nuovo comando IMP di sintassi

Rep \equiv **repeat** c_0 **break on** b **else** c_1 **end-repeat**

- Si dia la semantica operativa e denotazionale del nuovo comando, secondo la seguente semantica informale:

" c_0 viene eseguito nello stato attuale; se nello stato risultante l'espressione booleana b si valuta **TRUE** si esce, annullando l'esecuzione di c_0 , altrimenti si esegue c_1 (nello stato ottenuto dopo aver eseguito c_0) e si ritorna ad eseguire il comando **Rep**, nello stato ottenuto dopo aver eseguito c_1)

