Semantica operazionale e denotazionale

- Abbiamo definito la semantica di tre linguaggi di programmazione in termini operazionali, cioé in termini di sequenze di computazione

- Abbiamo assegnato un algoritmo che computa la funzione $\boldsymbol{\mathcal{M}}_{\boldsymbol{\mathcal{I}}}(S)$ (semantica operazionale) per ogni argomento (valore di input)

- La semantica denotazionale, $\boldsymbol{\mathcal{M}}_{\boldsymbol{\mathcal{I}}}(S)$, di un programma S, viene definita induttivamente

Introduzione alla semantica denotazionale

- La definizione induttiva di $\mathcal{M}_{\mathcal{I}}(S)$ è basata sulla definizione induttiva dell'insieme (sintattico) di tutti i programmi
- Problemi nella definizione di questa funzione possono essere causati da costrutti che portano a *loop*, come ad esempio while-statements e chiamate ricorsive

$$F(x) \Leftarrow \ \texttt{if}\ x = 0\ \texttt{then}\ 1\ \texttt{else}\ x * F(x-1)\ \texttt{fi}$$

viene considerato un'equazione in F, interpretando $\Leftarrow$ come il segno di eguaglianza

Il significato di questo programma è una soluzione in F a questa equazione

Introduzione alla semantica denotazionale (ctnd.1)

Per tali equazioni

1. si definisce cosa si intende per soluzione
2. ci si chiede se questa soluzione esiste e se è unica

Consideriamo

$F(x) \Leftarrow$ $\texttt{if}$ $x = 0$ $\texttt{then}$ 1 $\texttt{else}$ $x * F(x-1)$ $\texttt{fi}$

Una soluzione è una funzione monotona $f : Nat_\omega \rightarrow Nat_\omega$ tale che

$f(n) = \texttt{if-then-else}(n = 0, 1, n * f(n-1))$

per ogni $n \in Nat_\omega$

Introduzione alla semantica denotazionale(ctnd.2)

Queste funzioni sono esattamente i punti fissi del funzionale

$\Phi : (Nat_\omega \rightarrow_m Nat_\omega) \rightarrow (Nat_\omega \rightarrow Nat_\omega)$ definito da

$\Phi(f)(n) = \texttt{if-then-else}(n = 0, 1, n * f(n-1))$

CONSEGUENZA:

Il problema dell'esistenza e dell'unicità delle soluzioni di un programma ricorsivo viene ricondotto al problema dell'esistenza e dell'unicità dei punti fissi

Introduzione alla semantica denotazionale(ctnd.3)

Unicità?

$F(x) \Leftarrow F(x)$

$F(x) \Leftarrow \texttt{if } x = 0 \texttt{ then } F(x) \texttt{ else } 0 \texttt{ fi}$

NOTA:
Introdurremo delle proprietà che caratterizzano esattamente una soluzione

Ordini parziali piatti

Sia $S = \boldsymbol{\mathcal{D}}$ o $S = Bool$. Definiamo su S_ω la relazione $\sqsubseteq$ tale che dati $a, b \in S_\omega$, $a \sqsubseteq b$ se e solo se $a = \omega$ oppure $a = b$

Si legge: a è *meno definito di* b

$(S_\omega, \sqsubseteq)$ è un ordine parziale e viene chiamato *ordine parziale piatto* di S

Interpretazione per linguaggi di tipo $\mathcal{L}_3$

Sia $\mathcal{F}$ una base per $\mathcal{L}_3$. Un'*interpretazione* di $\mathcal{F}$ è una coppia $\boldsymbol{\mathcal{I}} = (\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{I}}_0)$ tale che $\boldsymbol{\mathcal{D}}$ è un insieme non vuoto detto *dominio*, e $\boldsymbol{\mathcal{I}}_0$ una funzione su $\mathcal{F}$ definita come segue

- Per ogni $c \in \mathcal{F}$, c di tipo b, $\boldsymbol{\mathcal{I}}_0(c) \in Bool$
- Per ogni $c \in \mathcal{F}$, c di tipo d, $\boldsymbol{\mathcal{I}}_0(c) \in \boldsymbol{\mathcal{D}}$
- Per ogni simbolo funzionale $f \in \mathcal{F}$ di tipo $(\beta_1, \ldots, \beta_n \rightarrow \beta_{n+1})$, dove $n \geq 1$ e $\beta_i \in \{b, d\}$, $\boldsymbol{\mathcal{I}}_0(f)$ è una funzione totale e monotona
  $\boldsymbol{\mathcal{I}}_0(f) : S_{1\omega} \times \ldots \times S_{n\omega} \rightarrow S_{(n+1)\omega}$
  dove $S_i = Bool$ se $\beta_i = b$ e $S_i = \boldsymbol{\mathcal{D}}$ se $\beta_i = d$, per $i = 1, \ldots, n+1$

Ordini parziali completi

Un ordine parziale $(D, \sqsubseteq)$ è *un ordine parziale completo (cpo)* se valgono le seguenti due condizioni

1 L'insieme D ha un elemento minimo. Tale elemento viene denotato con $\perp_D$ o semplicemente $\perp$

2 Per ogni catena S in D l'estremo superiore $\sqcup S$ esiste

Ogni ordine parziale che ha un elemento minimo e contiene solo catene finite, è un cpo

Ordini parziali completi (ctnd.)

Teorema
Se $(D_1, \sqsubseteq), \ldots, (D_n, \sqsubseteq)$ sono cpo, allora $(D_1 \times \ldots \times D_n, \sqsubseteq)$ è un cpo

CONSEGUENZA: $(Bool^2_\omega, \sqsubseteq)$, e $(Nat^2_\omega, \sqsubseteq)$ sono cpo

Teorema
Se D è un insieme e $(E, \sqsubseteq)$ un cpo, allora $((D \rightarrow E), \sqsubseteq)$ è un cpo

Ordini parziali completi - Esempio

$((Nat_\omega \rightarrow Nat_\omega), \sqsubseteq)$ è un cpo.

Un esempio di catena è l'insieme $S = \{f_i | i \in Nat\}$ dove $f_i : Nat_\omega \rightarrow Nat_\omega$ è definita

$$f_i(n) = \begin{cases} \omega & \text{se } n = \omega \text{ oppure } n \in Nat,\ n \geq i \\ \\ \text{n!} & \text{se } n \in Nat,\ 0 \leq n \leq i - 1 \end{cases}$$

L'estremo superiore di S è

$$(\sqcup S)(n) = \sqcup S(n) = \begin{cases} \omega & \text{se } n = \omega \\ \\ \text{n!} & \text{se } n \in Nat \end{cases}$$

Sotto-ordini parziali completi

Sia $(D, \sqsubseteq)$ un ordine parziale completo ed E un sottoinsieme di D. E viene chiamato un sotto-cpo di D se

1 l'ordine parziale $(E, \sqsubseteq)$ è un cpo

2 $\sqcup_E S = \sqcup_D S$ per ogni catena S in E

Caratterizzazione dei sotto-cpo

Sia $(D, \sqsubseteq)$ un cpo e sia E un sottoinsieme di D. Allora E è un sotto-cpo di D se e solo se valgono le seguenti due condizioni

1. E ha un elemento minimo

2. $\sqcup_D S$ sta in E per ogni catena S in E

Funzioni continue

Siano $(D, \sqsubseteq)$ e $(E, \sqsubseteq)$ cpo. Una funzione $f : D \longrightarrow E$ viene detta *continua* se per ogni catena S in D, l'estremo superiore $\sqcup f(S)$ esiste e $f(\sqcup S) = \sqcup f(S)$

L'insieme di tutte le funzioni continue da D in E viene denotato con $[D \longrightarrow E]$

Funzioni continue e funzioni monotone

Teorema

Siano $(D, \sqsubseteq)$ ed $(E, \sqsubseteq)$ due cpo, ed $f : D \longrightarrow E$ sia una funzione

1. La funzione f è continua se e solo se è monotona e per ogni catena S in D, $f(\sqcup S) \sqsubseteq \sqcup f(S)$
2. Se D contiene solo catene finite, allora f è continua se e solo se è monotona

Alcuni esempi

- Siano $D_{1\omega}, \ldots, D_{n\omega}, D_{(n+1)\omega}$ cpo piatti. Allora ogni funzione monotona $f : D_{1\omega} \times \ldots \times D_{n\omega} \longrightarrow D_{(n+1)\omega}$ è continua
- Ogni funzione $f : D \longrightarrow E$, tale che $f(d) = e$ per ogni $d \in D$ (costante) è continua
- La funzione identità $id : D \longrightarrow D$, tale che $id(d) = d$ per ogni $d \in D$ è continua
- Per $i = 1, \ldots, n$, la funzione proiezione $pr_i : D_1 \times \ldots \times D_n \longrightarrow D_i$ è continua

Alcuni esempi (ctnd. 1)

Consideriamo il funzionale
$\Phi : (Nat_\omega \longrightarrow Nat_\omega) \longrightarrow (Nat_\omega \longrightarrow Nat_\omega)$
definito da

$$\Phi(f)(n) = \boldsymbol{\mathcal{I}}(\texttt{if } x = 0 \texttt{ then } 1 \texttt{ else } x * F(x-1) \texttt{ fi})(\gamma[F/f][x/n])$$

$$\Phi(f)(n) = \begin{cases} \omega & \text{se } n = \omega \text{ oppure } n \neq 0 \text{ e } f(n-1) = \omega \\ \\ 1 & \text{se } n = 0 \\ \\ n * f(n-1) & \text{altrimenti} \end{cases}$$

Alcuni esempi (ctnd. 2)

Vogliamo dimostrare che il funzionale è continuo:

- Monotono: $f, g \in (Nat_\omega \to Nat_\omega)$. Se $f \sqsubseteq g$ allora $\Phi(f) \sqsubseteq \Phi(g)$

- data una catena S di funzioni in $(Nat_\omega \to Nat_\omega)$, $\Phi(\sqcup S) \sqsubseteq \sqcup \Phi(S)$

Alcuni esempi (ctnd. 3)

Scegliamo $f, g \in (Nat_\omega \rightarrow Nat_\omega)$ tali che $f \sqsubseteq g$. Vogliamo provare che per ogni $n \in Nat_\omega$, $\Phi(f)(n) \sqsubseteq \Phi(g)(n)$

- Se $n = \omega$, $\Phi(f)(n) = \omega$
- Se $n = 0$, $\Phi(f)(n) = 1 = \Phi(g)(n)$
- Se $n \neq 0, \omega$ e $f(n-1) = \omega$, allora $\Phi(f)(n) = \omega$
- Se $n \neq 0, \omega$ e $f(n-1) \neq \omega$, allora $f(n-1) = g(n-1)$ e dunque $\Phi(f)(n) = n * f(n-1) = n * g(n-1) = \Phi(g)(n)$

Alcuni esempi (ctnd. 4)

Sia S una catena in $(Nat_\omega \to Nat_\omega)$, per dimostrare che $\Phi(\sqcup S) \sqsubseteq \sqcup\Phi(S)$ si ponga $g = \sqcup S$. Dobbiamo dimostrare che per ogni $n \in Nat_\omega$
$\phi(g)(n) \sqsubseteq \sqcup\{\Phi(f)(n) | f \in S\}$

- Se $n = \omega$, $\Phi(g)(n) = \omega$
- Se $n = 0$, $\Phi(f)(n) = 1 = \Phi(g)(n)$ per ogni $f \in S$
- Se $n \neq 0, \omega$ e $f(n-1) = \omega$ per ogni $f \in S$, allora $g(n-1) = \omega$ e quindi $\Phi(g)(n) = \omega$
- Se $n \neq 0, \omega$ e $f(n-1) = m \neq \omega$ per qualche $f \in S$, allora $g(n-1) = m$ e dunque $\Phi(g)(n) = n * g(n-1) = n * m$. D'altra parte, $\sqcup\{\Phi(f)(n) | f \in S\} = \sqcup\{n * f(n-1) | f \in S\}$, è $n * m$.

Esempio: non tutte le funzioni monotone sono continue

$\Phi : (Nat_\omega \longrightarrow Nat_\omega) \longrightarrow Nat_\omega$

$$\Phi(f) = \begin{cases} 0 & \text{se } f(n) \neq \omega \text{ per tutti gli } n \in Nat \\ \\ \omega & \text{altrimenti} \end{cases}$$

Quando la funzione f è totale allora $\Phi(f) = 0$

Esempio: non tutte le funzioni monotone sono continue (ctnd.)

- Φ è monotona. Siano $f, g \in (Nat_\omega \rightarrow Nat_\omega)$ tali che $f \sqsubseteq g$

  – Se $\Phi(f) = \omega$, allora $\Phi(f) \sqsubseteq \Phi(g)$
  – Se $\Phi(f) = 0$, allora $\Phi(f) = \Phi(g)$

- Φ non è continua. Infatti è possibile esibire una catena di funzioni $S = \{f_i | i \in Nat\}$ dove $f_i : Nat_\omega \rightarrow Nat_\omega$ è definita

$$f_i(n) = \begin{cases} \omega & \text{se } n = \omega \text{ oppure } n \in Nat,\ n \geq i \\ \\ \text{n!} & \text{se } n \in Nat,\ 0 \leq n \leq i - 1 \end{cases}$$

tale che $\sqcup \Phi(S) = \omega \neq \Phi(\sqcup S) = 0$

Qualche proprietà (1)

Siano $(D, \sqsubseteq)$ ed $(E, \sqsubseteq)$ due cpo. Allora l'insieme $[D \longrightarrow E]$ di tutte le funzioni continue da D ad E è un sotto-cpo di $((D \longrightarrow E), \sqsubseteq)$

Siano $(D, \sqsubseteq)$, $(E, \sqsubseteq)$ ed $(F, \sqsubseteq)$ delle cpo e $g : D \longrightarrow E$ ed $f : E \longrightarrow F$ delle funzioni continue. Allora la loro composizione $f \circ g : D \longrightarrow F$ è pure continua

Qualche proprietà (2)

Siano $(D, \sqsubseteq)$, $(E_i, \sqsubseteq)$ per $i = 1, \ldots, n$ dei cpo, $n \geq 1$. Sia $f : D \longrightarrow E_1 \times \ldots \times E_n$ una funzione. Allora f è continua se e solo se $pr_i \circ f$ è continua per $i = 1, \ldots, n$ dove pr_i è l'i-esima funzione proiezione

Siano $(D, \sqsubseteq)$, $(E_i, \sqsubseteq)$ ed $(F, \sqsubseteq)$ delle cpo. Inoltre, sia $g_i \in [D \longrightarrow E_i]$ ed $f \in [E_1 \times \ldots \times E_n \longrightarrow F]$ funzioni continue, $i = 1, \ldots, n$, $n \geq 1$. La composizione di f e $g_1, \ldots, g_n$, cioé la funzione $h : D \rightarrow F$ definita da

$$h(d) = f(g_1(d), \ldots, g_n(d))$$

è continua

Minimo punto fisso

Sia $(D, \sqsubseteq)$ un ordine parziale ed $f : D \longrightarrow D$ una funzione. Il minimo punto fisso di f è l'elemento minimo dell'insieme di tutti i punti fissi di f

Il problema dell'esistenza del minimo punto fisso viene trattato analogamente a quello dell'esistenza dell'estremo superiore

Se il minimo punto fisso della funzione f esiste, viene indicato con μf

Teorema del punto fisso

Sia $(D, \sqsubseteq)$ un cpo ed $f : D \longrightarrow D$ una funzione continua. Allora f ha un minimo punto fisso μf e

$$\mu f = \sqcup\{f^i(\perp)|i \in Nat\}$$

dove $\perp$ è il minimo del cpo $(D, \sqsubseteq)$ ed f^i è l'i-esima iterazione della funzione f

Esempio

Consideriamo il funzionale
$\Phi : (Nat_\omega \longrightarrow Nat_\omega) \longrightarrow (Nat_\omega \longrightarrow Nat_\omega)$
definito da

$$\Phi(f)(n) =$$
$$\boldsymbol{\mathcal{I}}(\texttt{if } x = 0 \texttt{ then } 1 \texttt{ else } x * F(x-1) \texttt{ fi})(\gamma[F/f][x/n])$$

$$\Phi(f)(n) = \begin{cases} \omega & \text{se } n = \omega \text{ oppure } n \neq 0 \text{ e } f(n-1) = \omega \\ \\ 1 & \text{se } n = 0 \\ \\ n * f(n-1) & \text{altrimenti} \end{cases}$$

Ha un minimo punto fisso $\mu\Phi = \sqcup\{\Phi_i(\bot) | i \in Nat\}$

Esempio (ctnd.1)

Mostriamo che $\Phi^i(\perp) = f_i$, dove $f_i : Nat_\omega \rightarrow Nat_\omega$

$$f_i(n) = \begin{cases} \omega & \text{se } n = \omega \text{ oppure } n \in Nat,\ n \geq i \\ \\ \text{n!} & \text{se } n \in Nat,\ 0 \leq n \leq i-1 \end{cases}$$

CASO BASE: $\Phi^0(\perp) = \perp = f_0$

PASSO INDUTTIVO: Supponiamo che $\Phi^i(\perp) = f_i$. Si deve dimostrare che $\Phi^{i+1}(\perp) = f_{i+1}$.
Poiché $\Phi^{i+1}(\perp) = \Phi(f_i)$, basta far vedere che $\Phi(f_i) = f_{i+1}$

- Se $n = \omega$, allora $\Phi(f_i)(n) = \omega = f_{i+1}(n)$
- Se $n = 0$, allora $\Phi(f_i)(n) = 1 = 0! = f_{i+1}(n)$

- Se $n \neq 0, \omega$ e $n \leq i$, allora $f_i(n-1) = (n-1)! \neq \omega$. Ma allora $\Phi(f_i) = n * f_i(n-1) = n * (n-1)! = n! = f_{i+1}(n)$. Se invece $n \geq i+1$, allora $f_i(n-1) = \omega$, e quindi $\Phi(f_i)(n) = n * \omega = \omega = f_{i+1}(n)$.

Base per la λ-notazione

Una *base* per la λ-notazione è una tripla $B = (BT, F, V)$ di insiemi disgiunti di simboli dove:

- BT è un insieme di *tipi base*
- F è un insieme di *simboli funzionali*
- V è un insieme di *variabili*

Base per la λ-notazione(ctnd.)

Ad ogni simbolo funzionale e ad ogni variabile viene associato un tipo. L'insieme di tutti i tipi su BT viene definito induttivamente come segue:

- Ogni tipo base è un tipo
- se $\tau_1, \ldots \tau_n, \tau$ sono tipi $(n \geq 1)$, allora $(\tau_1, \ldots, \tau_n \to \tau)$ è un tipo

Si assume che l'insieme V contenga un numero infinito di variabili di ogni tipo.

Sintassi della λ-notazione

Data una base, l'insieme di λ-termini di ogni tipo sono definiti per induzione simultanea:

a. Ogni simbolo funzionale di tipo τ ed ogni variabile di tipo τ è un λ-termine di tipo τ

b. *Applicazione* Se $t_1, \ldots, t_n$ sono λ-termini di tipo $\tau_1, \ldots, \tau_n$ rispettivamente, $n \geq 1$, ed u è un λ-termine di tipo $(\tau_1, \ldots, \tau_n \rightarrow \tau)$, allora $u(t_1, \ldots, t_n)$ è un λ-termine di tipo τ.

   λ-*astrazione* Se $x_1, \ldots, x_n$ sono variabili distinte di tipo $\tau_1, \ldots, \tau_n$ rispettivamente, $n \geq 1$, e t è un λ-termine di tipo σ, allora $[\lambda x_1, \ldots, x_n.t]$ è un λ-termine di tipo $(\tau_1, \ldots, \tau_n \rightarrow \sigma)$

Interpretazione di una base per la λ-notazione

Sia $B = (BT, F, V)$ una base per la λ-notazione. Una *interpretazione* $\mathcal{I}$ di B è costituita da:

- Un cpo $(\mathcal{D}_\tau, \sqsubseteq)$ per ogni tipo base $\tau \in BT$
- Una funzione $\mathcal{I}_0$ che associa ad ogni simbolo funzionale $f \in F$ di tipo τ, un elemento $\mathcal{I}_0(f) \in \mathcal{D}_\tau$; quando τ non è un tipo base ma della forma $\tau = (\tau_1, \ldots, \tau_n \rightarrow \sigma)$, $\mathcal{D}_\tau$ è:
  $\mathcal{D}_\tau = [\mathcal{D}_{\tau_1} \times \ldots \times \mathcal{D}_{\tau_n} \rightarrow \mathcal{D}_\sigma]$

Interpretazione di una base per la λ-notazione (ctnd.)

Un *assegnamento* γ (per la base B e l'interpretazione $\boldsymbol{\mathcal{I}}$) è una funzione che mappa ogni variabile x di tipo τ in un elemento $\gamma(x) \in \boldsymbol{\mathcal{D}}_\tau$.

Sull'insieme degli assegnamenti Γ viene indotto l'ordine parziale: $\gamma \sqsubseteq \gamma'$ sse $\gamma(x) \sqsubseteq \gamma'(x)$ per tutte le variabili x in V

Semantica della λ-notazione

Per ogni interpretazione $\mathcal{I}$ di una base per la λ-notazione c'è un funzionale semantico, denotato anche $\mathcal{I}$, che mappa ogni λ-termine t di tipo τ in una funzione $\mathcal{I}(t) : \Gamma \to \mathcal{D}_\tau$ secondo la seguente definizione induttiva:

- $\mathcal{I}(f)(\gamma) = \mathcal{I}_0(f)$ se f è un simbolo funzionale
  $\mathcal{I}(x)(\gamma) = \gamma(x)$ se x è una variabile
- *Applicazione*
  $\mathcal{I}(u(t_1, \dots, t_n))(\gamma) = \mathcal{I}(u)(\gamma)(\mathcal{I}(t_1)(\gamma), \dots, \mathcal{I}(t_n)(\gamma))$
  λ-*astrazione* $\mathcal{I}([\lambda x_1, \dots, x_n.u])(\gamma) = f_\gamma$
  dove la funzione $f_\gamma : \mathcal{D}_{\tau_1} \times \dots \times \mathcal{D}_{\tau_n} \to \mathcal{D}_\sigma$ è definita

$$f_\gamma(d_1, \dots, d_n) = \mathcal{I}(u)(\gamma[x_1/d] \dots [x_n/d_n])$$

Semantica denotazionale di $\mathcal{L}_3$

Consideriamo il linguaggio di programmazione $\mathcal{L}_3$, con gli insiemi di variabili V e W e la base F. Definiamo una base $B = (BT, F', V')$ per la λ-notazione ponendo

- $BT = \{$b,d$\}$
- $F' = F$
- $V' = V \cup W$, dove ogni variabile $x \in V$ prende il tipo d ed ogni simbolo funzionale s-ario $F \in W$ prende il tipo $(d, \ldots, d \rightarrow d)$, dove $d, \ldots, d$ sta per una sequenza di s simboli d

Semantica denotazionale di $\mathcal{L}_3$ (ctnd.)

Sia $\boldsymbol{\mathcal{I}} = (\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{I}}_0)$ una interpretazione della base F per il linguaggio $\mathcal{L}_3$. Allora una interpretazione della base B per la λ-notazione viene creata facilmente:

- Gli cpo corrispondenti ai tipi base b e d sono $Bool_\omega$ e $\boldsymbol{\mathcal{D}}_\omega$
- L'interpretazione dei simboli funzionali in $F' = F$ viene data da $\boldsymbol{\mathcal{I}}_0$ ($\boldsymbol{\mathcal{I}}_0$ mappa i simboli funzionali in funzioni continue. Poiché in cpo piatti ci sono solo catene finite, queste funzioni sono continue)

Semantica denotazionale di $\mathcal{L}_3$ - Il funzionale semantico $\Phi(S)$

Sia $\boldsymbol{\mathcal{I}} = (\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{I}}_0)$ una interpretazione di una base per il linguaggio di programmazione $\mathcal{L}_3$ e sia S un programma ricorsivo definito dall'insieme delle equazioni ricorsive
$F_i(x_{i1}, \ldots, x_{is_i}) \Leftarrow t_i$
per $i = 1, \ldots, n$, $n \geq 1$ (e da un simbolo funzionale principale).
Si ponga
$\boldsymbol{\mathcal{D}}^* = [\boldsymbol{\mathcal{D}}_\omega^{s_1} \rightarrow \boldsymbol{\mathcal{D}}_\omega] \times \ldots \times [\boldsymbol{\mathcal{D}}_\omega^{s_n} \rightarrow \boldsymbol{\mathcal{D}}_\omega]$.
Il funzionale semantico $\Phi_{\boldsymbol{\mathcal{I}}}(S) : \boldsymbol{\mathcal{D}}^* \rightarrow \boldsymbol{\mathcal{D}}^*$ viene definito in ogni componente usando le funzioni proiezione come segue:

$$pr_1 \circ \Phi(S) = \boldsymbol{\mathcal{I}}([\lambda F_1, \ldots, F_n.[\lambda x_{11}, \ldots, x_{1s_1}.t_1]])(\gamma)$$
$$\vdots$$
$$pr_n \circ \Phi(S) = \boldsymbol{\mathcal{I}}([\lambda F_1, \ldots, F_n.[\lambda x_{n1}, \ldots, x_{ns_n}.t_n]])(\gamma)$$

dove γ è un assegnamento arbitrario

Semantica denotazionale di $\mathcal{L}_3$ - Il funzionale semantico $\Phi(S)$ (ctnd.)

Per il teorema del punto fisso (che può essere applicato poichè ogni $pr_i(\mu\Phi(S))$ è una funzione continua e D^* è un cpo):

$$\mu\Phi(S) = \sqcup\{(\Phi(S))^i(\perp, \ldots, \perp) | i \geq 0\}$$

Inoltre $pr_i(\mu\Phi(S)) \in [\boldsymbol{\mathcal{D}}_\omega^{s_i} \rightarrow \boldsymbol{\mathcal{D}}_\omega]$

Semantica denotazionale di $\mathcal{L}_3$ - Definizione

Sia S un programma ricorsivo su una base F definito dall'insieme delle equazioni ricorsive
$F_i(x_{i1}, \ldots, x_{is_i}) \Leftarrow t_i$
per $i = 1, \ldots, n$ e dalla variabile principale F_k, $1 \leq k \leq n$. Sia $\boldsymbol{\mathcal{I}} = (\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{I}}_0)$ una interpretazione di F e $\Phi(S)$ il funzionale semantico associato ad S e $\boldsymbol{\mathcal{I}}$. Il significato di S (nell'interpretazione $\boldsymbol{\mathcal{I}}$) è la funzione $\boldsymbol{\mathcal{M}}_{\boldsymbol{\mathcal{I}}}(S) : \boldsymbol{\mathcal{D}}^{s_k} \rightsquigarrow \boldsymbol{\mathcal{D}}$ definita da

$$\boldsymbol{\mathcal{M}}_{\boldsymbol{\mathcal{I}}}(S)(d_1, \ldots, d_{s_k}) = \begin{cases} pr_k(\mu\Phi_{\boldsymbol{\mathcal{I}}}(S))(d_1, \ldots, d_{s_k}) & \text{se questo valore è diverso da } \omega \\ \text{indefinito} & \text{altrimenti} \end{cases}$$