Fondamenti dei linguaggi di programmazione

Aniello Murano
Università degli Studi di Napoli
"Federico II"

Riassunto delle lezioni precedenti

- Prima Lezione: Introduzione e motivazioni del corso; Sintassi e semantica di ARITHM
- Seconda lezione: Sintassi e semantica operazionale del linguaggio imperativo IMP
- Terza lezione: Tecniche di prova per induzione (su IMP).
- Quinta lezione: Definizione induttiva di domini (per IMP).
- Occupiamoci adesso di semantica denotazionale.
- Prima introduciamo alcuni concetti matematici basilari:
  - Ordinamenti parziali completi, funzioni continue e minimi punti fissi
- Nella prossima lezione, mostreremo una semantica denotazionale per il linguaggio IMP e la sua equivalenza con la semantica operazionale.

Ordinamenti parziali completi, funzioni continue e minimi punti fissi

Introduzione

- La semantica operazionale mostra come possono essere utilizzati (eseguiti) i vari costrutti di linguaggio.
- Essa interpreta i programmi come diagrammi di transizione, con azioni che permettono di muoversi tra i vari stati.
- Queste strutture descrivono il comportamento di un programma, ma non permettono di definire completamente il suo significato (il significato è trasferito in un altro linguaggio).
- Potremmo definire una semantica operazionale come una formalizzazione matematica di qualche strategia di implementazione del programma.

Semantica denotazionale

- La semantica denotazionale cerca di catturare il significato interno di un programma piuttosto che la strategia di implementazione. Per questo motivo è più astratta di quella operazionale ed è indipendente dalla macchina su cui si lavora.
- Questa semantica mappa un linguaggio in un modello matematico astratto (invece di utilizzare regole operazionali), in modo tale che il valore di un programma composto è determinabile direttamente dai valori delle sue singole parti.
- In pratica, ad ogni frase del linguaggio viene associata una denotazione (significato) come funzione delle denotazioni delle sue sottofrasi.
- Ne consegue che in alcuni casi la semantica di una frase è il minimo punto fisso di una funzione opportunamente definita.

Strumenti matematici opportuni

- Lo spazio matematico solitamente utilizzato dalla teoria denotazionale della semantica dei linguaggi di programmazione è quello degli insiemi parzialmente ordinati (**domini**).
- Esempi di tali oggetti sono le **funzioni parziali**, unitamente ad un ordinamento parziale che permette di definire un concetto di approssimazione o di contenimento di informazione tra gli oggetti. Per esempio $f \leq g$ può indicare che g si comporta come f su tutti i valori in cui quest'ultima è definita.
- Esistono elementi del linguaggi la cui denotazione dipende dalla soluzione di equazioni ricorsive (come per il comando while di IMP). Vedremo che una soluzione per tali equazioni è data dal **minimo punto fisso** di un **insieme completo parzialmente ordinato**.

Ordinamento Parziale

- Un ordinamento parziale (p.o.) è una coppia (P, $\sqsubseteq$), dove P è un insieme di elementi e $\sqsubseteq$ è una relazione binaria su P che sia:
- **Riflessiva**: per ogni p in P, $p \sqsubseteq p$
- **Transitiva**: Per ogni p, q, r in P, se $p \sqsubseteq q \wedge q \sqsubseteq r$ allora $p \sqsubseteq r$
- **Antisimmetrica**: Per ogni p, q in P, se $p \sqsubseteq q \wedge q \sqsubseteq p$ allora p = q
- Esempi di insiemi parzialmente ordinati:
  - **(Z, $\leq$)**, dove Z è l'insieme dei numeri interi e $\leq$ è il solito ordinamento
  - **(2^S, $\subseteq$)**, dove **2^S** denota l'insieme delle parti di S, (spesso scritto come $\mathcal{P}$**(S)**, o $\mathcal{P}$**ow(S)**
- Esempi di insiemi non parzialmente ordinati:
  - **(Z, <)** non è un p.o. perché **...**
  - **(Z, $\sqsubseteq$)**, dove **m** $\sqsubseteq$ **n** $\Leftrightarrow$ **|m|** $\leq$ **|n|**, non è un p.o. perché **...**

Ordinamento Parziale Completo

- Dato un ordinamento parziale (P, $\sqsubseteq$) e un sottoinsieme $X \subseteq P$, allora p è un **upper bound** (maggiorante) di X sse $\forall q \in X$, $q \sqsubseteq p$.
- Si dice che p è un **least upper bound** (minimo maggiorante, **lub**) di X sse
  - p è un upper bound di X
  - Per ogni upper bound q di X, $p \sqsubseteq q$

1. Un lub di un sottoinsieme **X** di un p.o. è anche indicato con $\bigsqcup$**X**

- Una **catena** di un p.o. (P, $\sqsubseteq$), è una sequenza $d_0 \sqsubseteq d_1 \sqsubseteq ... \sqsubseteq d_n$, il cui lub è d_n (un insieme finito totalmente ordinato è una catena).
- Una ω**-catena** di un p.o. è una catena infinita. Anch'essa può avere un lub e sarà indicato con $\bigsqcup_{n \in \omega} d_n$ (intuitivamente, l'elemento limite della catena
- Un p.o. (P, $\sqsubseteq$) è un **ordinamento parziale completo**(c.p.o.) se tutte le sue ω-catene hanno un lub
- Un c.p.o. (P, $\sqsubseteq$) è con **elemento minimo** se esiste un elemento $\perp \in P$ tale che $\forall q \in P$, $\perp \sqsubseteq q$

Esempi di c.p.o.

- **(2^S, $\subseteq$)** è un c.p.o., ed il minimo upper bound (lub) per la catena di tutti gli elementi di 2^S è proprio S ($\bigcup = \sqcup$).
- **($\mathcal{N}$, $\leq$)** non è un c.p.o. perchè $\mathcal{N}$ non ha un lub. Se invece consideriamo ($\mathcal{N} \cup \infty$, $\leq$), questo è un c.p.o dove il lub è ∞.
- **([0,1], $\leq$)** è un c.p.o. dove [0,1] è l'intervallo continuo chiuso di reali e 1 è un lub per catene infinite. Si noti come ([0,1), $\leq$) non è un c.p.o.. Infatti, non esiste un lub per la catena infinita, 1/2, 2/3, 3/4… che ha per limite 1 $\notin$ [0,1)
- Se (P, $\sqsubseteq$) è un c.p.o. lo è anche (P, $\sqsupseteq$)?
- No, perché (P, $\sqsupseteq$) potrebbe non avere un lub. Per esempio, ((0,1], $\leq$) è un c.p.o. con lub 1, mentre ((0,1], $\geq$) non ha un lub. Infatti, non esiste un lub per la catena infinita, 1/2, 1/3, 1/4… che ha per limite 0 $\notin$ (0,1]

Funzioni monotone e continue

- Siano (D, $\sqsubseteq$) e (E, $\sqsubseteq$) due c.p.o. Una funzione f: D $\rightarrow$ E si dice monotona sse preserva il seguente ordine:

$$\forall\, d,d' \in D.\ d \sqsubseteq d' \Rightarrow f(d) \sqsubseteq f(d')$$

- La funzione f si dice continua se e solo se è monotona e per ogni catena $d_0 \sqsubseteq d_1 \sqsubseteq \ldots \sqsubseteq d_n \sqsubseteq \ldots$ contenuta in D vale

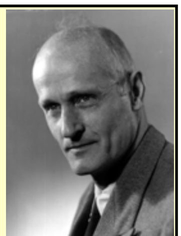
$$\bigsqcup_{n\in\omega} f(d_n) = f(\bigsqcup_{n\in\omega} d_n)$$

- Come vedremo, questa formulazione è molto utile per calcolare il minimo punto fisso di una funzione

Punto fisso

- Sia (D, $\sqsubseteq$) un c.p.o. e f: D $\rightarrow$ D una funzione continua. Un **punto fisso** di f è un elemento d $\in$ D tale che f(d) =d.
  - La funzione polinomiale sui numeri reali $f(x) = x^2 - 3x + 4$ ha punto fisso e precisamente f(2) = 2.
  - La funzione f(x)=x+1 non ha punto fisso sui reali perché x non è mai uguale a x+1, per qualsiasi numero reale x.
- Un **pre-punto** fisso di f è un elemento d $\in$ D tale che f(d) $\sqsubseteq$ d.
- Una funzione può avere più punti fissi
- **Thm[Kleene]: Una funzione continua su un c.p.o. ha sempre un punto fisso.**
- Per essere precisi, anche una funzione **monotona** su un c.p.o ammette punto fisso, ma se la funzione è anche continua allora abbiamo un algoritmo per calcolare il punto fisso, come vedremo nelle prossime diapositive.

Intermezzo
Stephen Cole Kleene [1909-1994]



- Kleene è un matematico americano.
- Nato a Hartford, Connecticut(USA), si laureò nel 1930.
- Dal 1930 al 1935, fu assistente ricercatore all'università di Princeton, dove ricevette il dottorato in matematica nel 1934, supervisionato da Alonzo Church.
- Da 1935 lavorò all' università di Winsconsin-Madison dove predispose le fondamenta dell'informatica teorica.
- Kleene è noto per la fondazione del ramo della logica matematica conosciuta come **teoria della ricorsione**, insieme con Alonzo Church, Kurt Godel, Alan Turing ed altri, e per aver inventato le **espressioni regolari**.
- Fornendo metodi per determinare quali problemi sono risolvibili, il suo lavoro portò allo studio di quali funzioni fossero **calcolabili**.
- Tra le altre cose, l'algebre di Kleene, la star di Kleene, il teorema di ricorsione di Kleene ed il teorema di **punto fisso** di Kleene sono stati chiamati così in suo onore.

Terema del Punto fisso (1)

- Sia (D, $\sqsubseteq$) un c.p.o. con minimo $\perp$ e f: D $\to$ D una funzione continua. Allora si ha che:
  - $\perp \sqsubseteq f(\perp)$ perché $\perp \sqsubseteq x$ per ogni elemento $x \in D$ (e di f(D))
  - $f(\perp) \sqsubseteq f(f(\perp))$ perché f è monotona e vale la riga precedente.
  - $f(f(\perp)) \sqsubseteq f(f(f(\perp)))$ per lo stesso motivo di prima e così via
- Sia $f^{n+1}(\perp) = f(f^n(\perp))$.
- $\perp \sqsubseteq \mathbf{f(\perp)} \sqsubseteq \mathbf{f^2(\perp)} \sqsubseteq \mathbf{f^3(\perp)}$... è una catena in D
- Sia $fix(f) = \bigsqcup_{n\in\omega} f^n(\perp)$allora
- **Teorema:** fix(f) è il minimo punto fisso di f, cioè, fix(f) è un punto fisso di f ed è il suo minimo pre-punto fisso (cioè, il minimo punto fisso di f è il least upper bound della catena).
  - In simboli: **(1) f(fix(f))=fix(f) e (2) se f(d) $\sqsubseteq$ d allora fix(f) $\sqsubseteq$ d**

Teorema di punto fisso (2)

- Proviamo il primo punto!
- Ricordiamo che $fix(f) = \bigsqcup_{n\in\omega} f^n(\perp)$
- Per la continuità di f abbiamo che

$$\begin{aligned} f(fix(f)) &= f(\bigsqcup_{n\in\omega} f^n(\perp)) \\ &= (\bigsqcup_{n\in\omega} f(f^n(\perp))) \\ &= (\bigsqcup_{n\in\omega} f^{n+1}(\perp)) \\ &= (\bigsqcup_{n\in\omega} f^{n+1}(\perp)) \sqcup \{\perp\} \\ &= (\bigsqcup_{n\in\omega} f^n(\perp)) = fix(f) \end{aligned}$$

È la catena senza il primo elemento

Se aggiungo bottom, il risultato non cambia e ottengo la stessa catena di partenza

Dunque fix(f) è un punto fisso

Teorema di punto fisso (3)

- Proviamo il secondo punto!
- Sia d un pre-punto fisso. Allora $\perp \sqsubseteq d$ (per definizione di bottom).
- Per la monotonicità di f, $f(\perp) \sqsubseteq f(d)$
- Siccome d è un pre-punto fisso, per definizione $f(d) \sqsubseteq d$. Dunque $f(\perp) \sqsubseteq d$.
- Per induzione $f(f(\perp)) \sqsubseteq f(d) \sqsubseteq d$.... Dunque $f^n(\perp) \sqsubseteq d$.
- Ma allora

$$fix(f) = (\bigsqcup_{n \in \omega} f^n(\perp)) \sqsubseteq d$$

- **Dunque fix(f) è il minimo punto fisso di f**

Osservazioni

- A questo punto abbiamo acquisito gli strumenti matematici di base per discutere della semantica denotazionale dei linguaggi di programmazione.
- In che modo questi strumenti vengono utilizzati?
- Gli ordinamenti parziali completi corrispondono ai tipi di dati (sia in input che output) di una computazione
- Le funzioni calcolabili sono rappresentate da funzioni continue fra c.p.o.
- Gli elementi di un c.p.o sono punti di informazione
- $x \sqsubseteq y$ può essere interpretato come "x approssima y" (oppure "x ha meno informazioni di y"). Di conseguenza $\perp$ è il punto con minima informazione.