∈∉Φπσδη↔←→¬∨∧⊆≠∪

Lezione 9

DCL: Programmi in Tipi di Dati Astratti.
Esempi

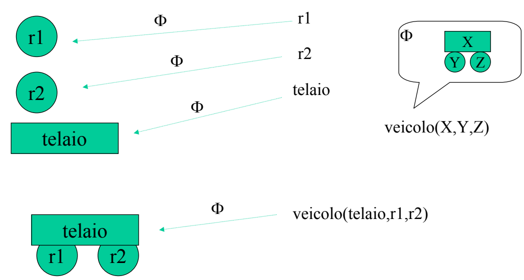
Dalla lezione scorsa

- Abbiamo visto l'uso di DCL con costanti come:
  - linguaggio per basi dati
  - RRS
- Abbiamo accennato all'uso di funzioni.
- Vediamo oggi l'uso di funzioni nella assiomatizzazione di tipi di dati.
- Vediamo poi degli esempi.

1. Interpretazioni con funzioni

- I = (D, Φ, π) come in precedenza, ma ora Φ interpreta anche i simboli di funzione:
  - c si interpreta in $c^{\Phi} \in D$
  - f/n si interpreta in $f^{\Phi} : D^n \rightarrow D$

Esempio



1.1. DOMINIO CHIUSO E UNIVERSO DI HERBRANDT

- L'esempio rappresenta il punto di vista da usare in DCL:
  - costanti: ***dati semplici***, non strutturati
  - termini ground (non costanti): ***dati strutturati***
- termine ground come dato: rappresentazione astratta di un individuo, oggetto concreto o astratto
- l'insieme dei termini ground è detto ***universo di Herbrandt*** U.
- NOTA. Un universo U si introduce mediante una ***segnatura,*** costituita da insieme di simboli di
  - funzione n-aria f/n
  - caso particolare: costante c/0

Esempio

- Segnatura: v/0, f/0, nil/0, cons/2
- Universo generato:

  profondità 0:

  v, f, nil,

  profondità 1:

  cons(v,v), cons(f,f), cons(f,v), cons(v,f), cons(nil,f), ….

  profondità 2:

  cons(v,cons(f,f)), cons(cons(f,v),cons(v,v)), …

  …….

1.2. La segnatura di una base di conoscenza KB; universo e base di Herbrandt

- La segnatura di una KB è
  - l'insieme dei simboli di funzione e costante usati in KB;
    - genera ***l'universo di Herbrandt*** U_{KB}
  - l'insieme dei simboli di predicato usati in KB;
    - genera la ***base di Herbrandt*** B_{KB}
- Possiamo fare diverse ipotesi circa l'universo:

- ***Ipotesi del dominio chiuso:***
  - tutti gli individui sono rappresentati da termini di U_{KB}
- ***Ipotesi del dominio aperto:***
  - non conosciamo a priori l'insieme di tutti gli individui; alcuni non hanno denotazione in U_{KB}
- ***Ipotesi del nome unico:***
  - termini ground distinti rappresentano individui diversi.

Esercizio

- Si consideri il programma Vocali:

  vocale(a).
  vocale(e).
  vocale(i).
  vocale(o).
  vocale(u).
  consonante(d).
- Qual è la segnatura di Vocali?
- Qual è l'universo di Herbrandt?
- Pensando all'alfabeto, ha senso l'ipotesi del nome unico?
- E quella del dominio chiuso?

2. Interpretazioni e modelli di Herbrandt

- Nell'ipotesi di dominio chiuso e nome unico:
- ***DEF. Interpretazione di Herbrandt*** $H \subseteq B_{KB}$
- Ricordiamo la nozione di modello:

  H ***modello di Herbrandt*** di KB, scritto H |= KB, sse
  H |= C per ogni clausola C di KB

NOTA: si ricordi la definizione di verità per clausole aperte:
H |= C sse H |= $C\sigma$ per ogni istanza ground $C\sigma$ di C
(σ sostituisce ogni variabile con un termine ground di U_{KB})

- Come già visto per Datalog, esiste sempre il ***minimo modello di Herbrandt***
  - un atomo A appartiene al minimo modello di KB sse appartiene a tutti i modelli
  - un atomo A appartiene al minimo modello di KB sse KB |= A (A è conseguenza logica di KB)
  - un atomo A appartiene al minimo modello di KB sse KB |- A (esiste una prova di A da KB)

Esempio

- Considerando la segnatura v/0, f/0, nil/0, cons/2,
- vogliamo isolare i termini dell'universo che rappresentano
  - booleani: v ed f
  - liste di booleani: es.: lista v,v,f,f
- Introduciamo i predicati
  - bool/1, listBool/1
- la base di Herbrandt sarà:
  - bool(v), bool(f), bool(nil), bool(cons(v,v)), list(v), list(f), list(nil), ………

- *Decidiamo* di rappresentare i booleani con
  - v, f
- e una lista di booleani x1,x2,..,xn con
  - cons(x1,cons(x2,cons(…..cons(xn,nil)….)))
  - x1 *testa;* cons(x2,cons(…..cons(xn,nil)….)) *coda*
- L'interpretazione di Herbrandt che rappresenta la nostra decisione è
  - bool(v), bool(f),
  - listBool(nil),
  - listBool(cons(v,nil)), listBool(cons(f,nil)),
  - listBool(cons(v,cons(v,nil))), listBool(cons(f, cons(v,nil))), …

- Per le liste si ha un insieme infinito di atomi. Come caratterizzarlo rigorosamente? Con una ***definizione induttiva:***
  - definiamo le liste di profondità 0
  - indichiamo come, a partire da liste di profondità i, passiamo a liste di profondità i+1

listBool(nil).
listBool(cons(T,C)) :- bool(T), listBool(C).
bool(v).
bool(f).

- Il ***modello minimo*** di listBool è esattamente l'interpretazione di Herbrandt che elenca le liste di booleani.

3. Le procedure bottom-up e top-down

- La procedura bottom-up lavora con il solito operatore T_{KB} basato sulla conseguenza immediata, dove, data una KB ed un insieme H di atomi: $A \in T_{KB}(H)$ sse
  - $A \in H$
  - A è un'istanza ground di un fatto di KB
  - esiste un'istanza ground $A \leftarrow$ Body di una clausola di KB e gli atomi del Body appartengono ad H
- La procedura top-down lavora esattamente nel modo visto in precedenza per Datalog, usando l'unificazione.

Esempio bottom-up

bool(v).
bool(f).
listBool(nil).
listBool(cons(T,C)) :- bool(T), listBool(C).

$T_{KB}(\emptyset)$ = I1 = {bool(v), bool(f), listBool(nil)}
T_{KB}(I1) = I2 = {bool(v), bool(f), listBool(nil), listBool(cons(v,nil)), listBool(cons(f,nil))}
T_{KB}(I2) = I3 = {bool(v), bool(f), listBool(nil), listBool(cons(v,nil)), listBool(cons(f,nil)), listBool(cons(v, cons(v,nil))), listBool(cons(v, cons(f,nil))), listBool(cons(f, cons(v,nil))), listBool(cons(f, cons(f,nil)))}
………………