

FONDAMENTI DI INTELLIGENZA ARTIFICIALE

PARTE II

15 Giugno 2010 – Tempo a disposizione 45minuti – Risultato 32/32 punti

Esercizio 1 (punti 16)

Si scriva un metainterprete per Prolog che all'interno del body di ogni clausola selezioni sempre prima quei sottogoal con un minor numero di parametri.

Ad esempio, nel body di

$p(X, Y, Z) :- b(Y, Y), n, a(X, Y), c(Z).$

selezionerà prima n , poi $c(Z)$ e poi, indifferentemente $a(X, Y)$ e $b(Y, Y)$

Si supponga di avere a disposizione un predicato $sort(L, L1, L2)$ che fornisce in $L2$ la lista L ordinata (in senso decrescente) secondo i valori contenuti in $L1$.

Si supponga, inoltre, di avere un programma nella forma $clausola(Head, Body)$ dove $Body$ è una lista di sottogoal.

Esercizio 2 (punti 6)

Si supponga di utilizzare un semplice sistema di produzioni (basato su regole forward) per ordinare una stringa fatta di "a", "b" e "c".

▪ *Insieme di regole:*

1. $ba \Rightarrow ab$
2. $ca \Rightarrow ac$
3. $cb \Rightarrow bc$

Le regole vengono utilizzate nel seguente modo: si considera la stringa attualmente nella memoria di lavoro, e si considerano applicabili tutte le regole che hanno come preconditione una sottostringa contenuta in tale stringa. Ad esempio, se nella memoria di lavoro è presente la stringa "cbad", si possono applicare le regole (1) e (3). Si sceglie una regola da applicare secondo una strategia (vedi sotto): l'applicazione di una regola produce una nuova stringa, in cui la sottostringa presente nella preconditione della regola viene riscritta come specificato dalla regola. La nuova stringa così prodotta sostituisce nella memoria di lavoro la stringa precedente. Ad esempio, supponendo di applicare la regola (1) alla stringa "cbad", si ottiene la nuova stringa "cabd", che va a sostituire la stringa precedente.

- *Memoria di lavoro:* la stringa in una fase intermedia del processo di ordinamento. All'inizio la stringa da ordinare è "cbaca".
- *Strategia:* la prima regola applicabile

Si descrivano i passaggi che portano alla soluzione riempiendo opportunamente la seguente tabella:

Ciclo	Memoria di lavoro	Insieme dei conflitti	Regola selezionata

Esercizio 3 (punti 5)

Si spieghi cosa si intende per “frame” problem nella pianificazione e si riportino come esempio in Prolog alcuni degli assiomi che lo trattano nella formulazione (calcolo delle situazioni) del mondo a blocchi di Kowalski.

Esercizio 4 (punti 5)

Nell'ambito dello standard OWL basato su Description Logics, descrivere brevemente cosa si intende quando una relazione (proprietà) è definita come "funzionale". Aggiungere un brevissimo esempio esplicativo.

Soluzioni:

Soluzione
Esercizio 1

```

solve([]).
solve([Head|Tail]):-!,
    solve(Head),
    solve(Tail).
solve(Goal):-
    clausola(Goal, Body),
    ordina_body(Body, Body_ord),
    solve(Body_ord).

ordina_body(Body, Body_ord):-
    numero_par(Body, Parametri),
    sort(Body, Parametri, Body_rev),
    reverse(Body_rev, Body_ord).

numero_par([], []).
numero_par([Head|Tail],[Nhead |Ntail]):-
    Head=..[_|Arg],
    conta(Arg, Nhead),
    numero_par(Tail, Ntail).

conta([], 0):- !.
conta([Arg|Tail], Nout):-
    conta(Tail, Nin), Nout is Nin +1.

```

Esercizio 2

Soluzione

Ciclo	Memoria lavoro	di Insieme conflitti	dei Regola selezionata
0	cbaca	1, 2, 3	1
1	cabca	2	2
2	acbca	2, 3	2
3	acbac	1, 3	1
4	acabc	2	2
5	aacbc	3	3
6	aabcc	0	ALT