

COMPITO DI INTELLIGENZA ARTIFICIALE

Fondamenti di Intelligenza Artificiale e Intelligenza Artificiale v.o. – parte I - Modulo A

22 Giugno 2006 - Tempo 2 ore

Esercizio 1 (punti 7)

L'algoritmo di Euclide per il calcolo del massimo comun divisore è dato dalla seguente definizione:

$$\text{mcd}(A, B) = \begin{cases} A & \text{se } A = B \\ \text{mcd}(A, B - A) & \text{se } A < B \\ \text{mcd}(A - B, B) & \text{se } A > B \end{cases}$$

Si consideri il seguente programma Prolog, che è l'implementazione dell'algoritmo di Euclide

```
mcd(A,A,A):-!.
mcd(A,B,M):- A<B,!, C is B-A, mcd(A,C,M).
mcd(A,B,M):- I is A-B, mcd(I,B,M).
```

Sebbene sembri corretto, l'algoritmo va in loop con il goal

```
?- mcd(2,2,1).
```

Si corregga il programma e si disegni l'albero SLD relativo all'invocazione

```
?- member(X,[1,2]), mcd(2,X,1).
```

dove il predicato member/2 è definito da

```
member(Y,[Y|_]).
member(Y,[_|_]):- member(Y,_).
```

Esercizio 2 (punti 5)

Si definisca un predicato `prodotto_cartesiano(L1,L2,L3)` che, date due liste qualsiasi L1 e L2 restituisca la lista L3 delle coppie ordinate che si possono formare con elementi di L1 e L2. Per esempio, se L1=[a,b,c], L2 = [a,d] la lista risultante deve essere L3=[[a,a],[a,d],[b,a],[b,d],[c,a],[c,d]]. Si definiscano tutti i predicati utilizzati, anche se già visti a lezione.

Esercizio 3 (punti 7)

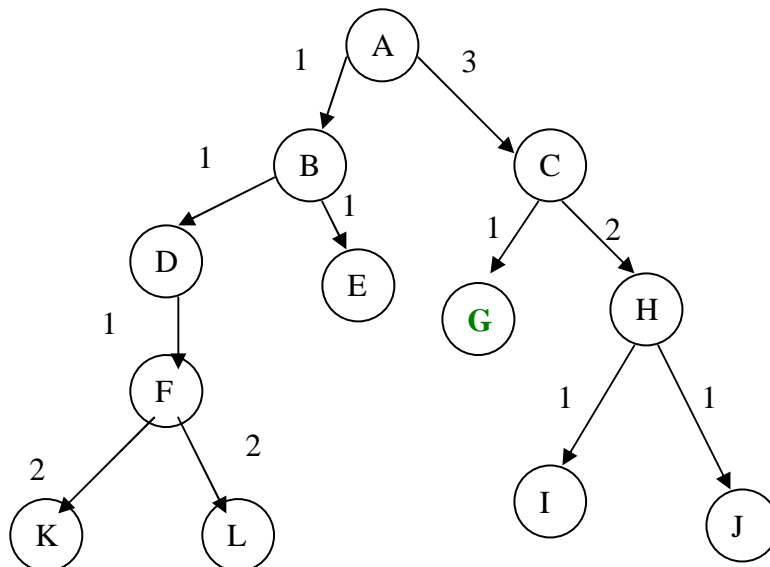
Si formalizzino le seguenti frasi in logica dei predicati:

- Giorgio e Paolo amano le stesse donne
- Se Giorgio ama una donna allora è felice.
- Giorgio non è felice.

Le si trasformi in clausole, si indichi se sono tutte clausole di Horn e poi si usi poi il principio di risoluzione per derivare il teorema "Paolo non ama alcuna donna".

Esercizio 4 (punti 5)

Si consideri il seguente albero di ricerca in cui i numeri sugli archi sono i costi.



Si assuma che i nodi siano espansi in ordine alfabetico, nel caso in cui la strategia di ricerca applicata li consideri equivalenti.

Il goal sia lo stato G e la ricerca termini al suo raggiungimento.

Si indichi la lista di stati espansi da ognuna delle seguenti strategie di ricerca:

Breadth First

Depth First

Iterative Deepening Search

Uniform Cost Search

Si indichi inoltre complessità Spaziale, Temporale, completezza ed ottimalità di ognuna di tali strategie, motivando la risposta.

Esercizio 5 (punti 8)

Si modelli il seguente problema a vincoli: si hanno 6 attività di cui devo stabilire una posizione relativa rispetto alle altre. Sappiamo che:

- L'attività A non è la prima e non è l'ultima.
- L'attività C è dopo l'attività B (non necessariamente contigue)
- L'attività E sta tra l'attività A e l'attività C
- L'attività F viene subito dopo l'attività A
- Le attività non possono sovrapporsi

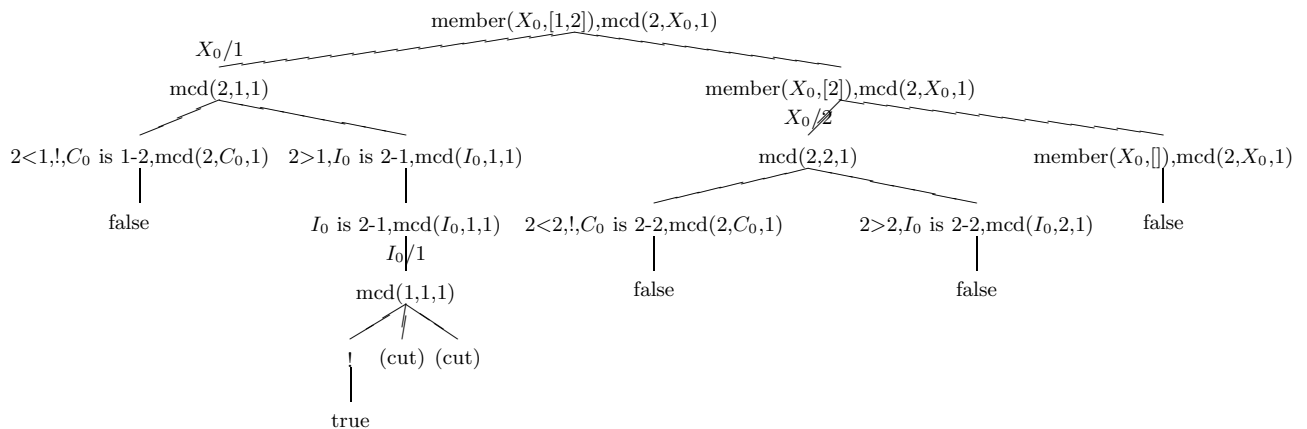
Si modelli il problema usando opportune variabili domini e vincoli. Si applichi l'arc consistenza al problema originale. Si risolva il problema mostrando l'albero fino alla prima soluzione utilizzando l'arc consistenza ad ogni nodo e selezionando le variabili con l'euristica MRV (minimum remaining values). In caso di parità con euristica MRV si selezionino le variabili in ordine alfabetico, dalla prima all'ultima.

SOLUZIONE

Esercizio 1

Per evitare che il programma vada in loop, bisogna evitare che venga selezionata la terza clausola quando $A=B$. Una soluzione è quindi imporre che la terza clausola venga selezionata solo se $A>B$.

```
mcd(A,A,A):-!.
mcd(A,B,M):- A<B,!, C is B-A, mcd(A,C,M).
mcd(A,B,M):- A>B, I is A-B, mcd(I,B,M).
```



Un'alternativa è quella di far sì che venga selezionata la prima clausola in tutti i casi in cui i due operandi sono uguali ed effettuare a posteriori (dopo il cut) l'unificazione del risultato con A:

```
mcd(A,A,M):-!, A=M.
mcd(A,B,M):- A<B,!, C is B-A, mcd(A,C,M).
mcd(A,B,M):- I is A-B, mcd(I,B,M).
```

Esercizio 2

```
prodotto_cartesiano([],_,[]).
prodotto_cartesiano([A|T],L2,L3):-
    prod(A,L2,L), prodotto_cartesiano(T,L2,L4), append(L,L4,L3).
```

```
prod(A,[],[]).
prod(A,[H|T],[[A,H]|M]):- prod(A,T,M).
```

```
append([],X,X).
append([X|L1],L2,[X|L3]):-
    append(L1,L2,L3).
```

Esercizio 3

Traduzione in logica

- $\forall X(\text{donna}(X) \Rightarrow (\text{ama}(\text{giorgio}, X) \Leftrightarrow \text{ama}(\text{paolo}, X)))$
 - $\forall X(\text{donna}(X) \wedge \text{ama}(\text{giorgio}, X) \Rightarrow \text{felice}(\text{giorgio}))$
 - $\neg \text{felice}(\text{giorgio})$
- G: $\forall X(\text{donna}(X) \Rightarrow \neg \text{ama}(\text{paolo}, X))$

Trasformazione in clausole

- $\neg donna(X) \vee \neg ama(giorgio, X) \vee ama(paolo, X)$
- $\neg donna(X) \vee ama(giorgio, X) \vee \neg ama(paolo, X)$
- $\neg donna(X) \vee \neg ama(giorgio, X) \vee felice(giorgio)$
- $\neg felice(giorgio)$
- $donna(c1)$
- $ama(paolo, c1)$

Risoluzione

1. $\neg donna(X) \vee \neg ama(giorgio, X) \vee ama(paolo, X)$
2. $\neg donna(X) \vee ama(giorgio, X) \vee \neg ama(paolo, X)$
3. $\neg donna(X) \vee \neg ama(giorgio, X) \vee felice(giorgio)$
4. $\neg felice(giorgio)$
5. $donna(c1)$
6. $ama(paolo, c1)$
7. (da 6, 2, 5) $ama(giorgio, c1)$
8. (da 7, 3, 5) $felice(giorgio)$
9. (da 8, 4) []

Esercizio 4

Breadth First	A B C D E G
Depth First	A B D F K L E C G
Iterative Deepening Search	A A B C A B D E C G
Uniform Cost Search	A B D E C F G

Per complessità Spaziale, Temporale, completezza ed ottimalità di ognuna di tali strategie si consultino le dispense.

Esercizio 5

Variabili e Domini A..F :: [1..6]

Vincoli

$A \neq 1, A \neq 6$

$C > B$

$E > A, C > E$

$A + 1 = F$

$A \neq B, A \neq C \dots E \neq F$

Arc consistenza:

A::[2..4]

B::[1..5]

C::[4..6]

D::[1..6]

E::[3..5]

F::[3..5]

Prendo una variabile con dominio di cardinalità minore $A = 2$ e riapplico l'arc consistenza

$B::[1, 4, 5]$

$C::[5, 6]$

$D::[1, 4..6]$

$E::[4, 5]$

$F = 3$

Prendo una variabile con dominio di cardinalità minore $C = 5$ e riapplico l'arc consistenza

$A = 2$

$B = 1$

$D = 6$

$C = 5$

$E = 4$

$F = 3$