

FONDAMENTI DI INTELLIGENZA ARTIFICIALE – 1° parte

21 Dicembre 2011 – Soluzioni

Esercizio 1

I lettura (eccetto “debole”)

$\forall x \text{ Francese}(x) \wedge \neg \text{Parigino}(x) \Rightarrow \text{Gentile}(x)$

Parigino (Jacques)

$\neg \text{Gentile}(\text{Jacques}) \vee \neg \text{Francese}(\text{Jacques})$

Goal negato: $\text{Gentile}(\text{Jacques}) \wedge \text{Francese}(\text{Jacques})$

In forma a clausole:

1. { $\neg \text{Francese}(x)$, $\text{Parigino}(x)$, $\text{Gentile}(x)$ }
2. { $\text{Parigino}(\text{Jacques})$ }
3. { $\text{Gentile}(\text{Jacques})$ }
4. { $\text{Francese}(\text{Jacques})$ }

5. { $\text{Parigino}(\text{Jacques})$, $\text{Gentile}(\text{Jacques})$ } da 4 e 1

ma poi non si riesce a dedurre altro, quindi non è conseguenza logica.

II lettura (eccetto “forte”)

$\forall x (\text{Francese}(x) \wedge \neg \text{Parigino}(x) \Rightarrow \text{Gentile}(x)) \wedge (\text{Parigino}(x) \Rightarrow \neg \text{Gentile}(x))$

Parigino (Jacques)

$\neg \text{Gentile}(\text{Jacques}) \vee \neg \text{Francese}(\text{Jacques})$

Goal negato: $\text{Gentile}(\text{Jacques}) \wedge \text{Francese}(\text{Jacques})$

In forma a clausole:

1. { $\neg \text{Francese}(x)$, $\text{Parigino}(x)$, $\text{Gentile}(x)$ }
2. { $\neg \text{Parigino}(x)$, $\neg \text{Gentile}(x)$ }
3. { $\text{Parigino}(\text{Jacques})$ }
4. { $\text{Gentile}(\text{Jacques})$ }
5. { $\text{Francese}(\text{Jacques})$ }

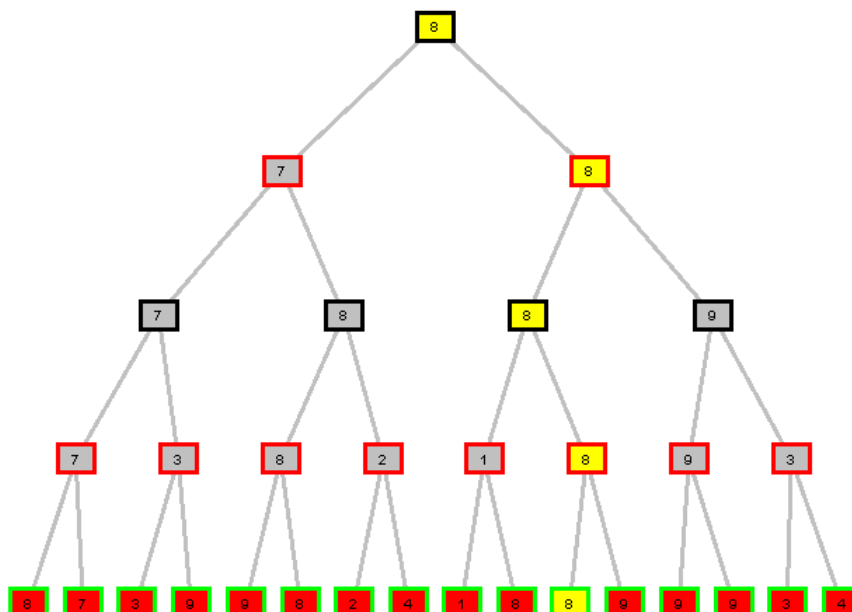
6. { $\neg \text{Parigino}(\text{Jacques})$ } da 4 e 2

7. { } da 6 e 3

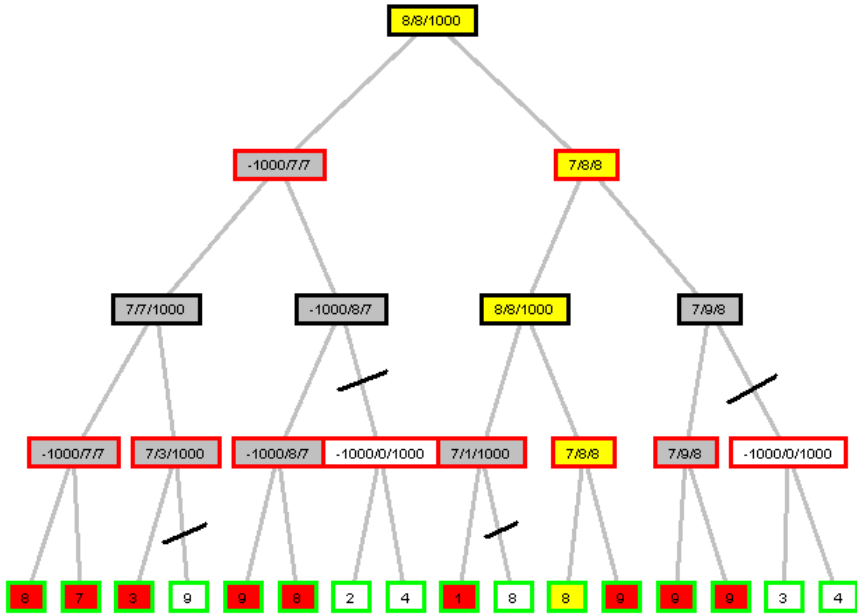
quindi è conseguenza logica.

Esercizio 2

min-max:



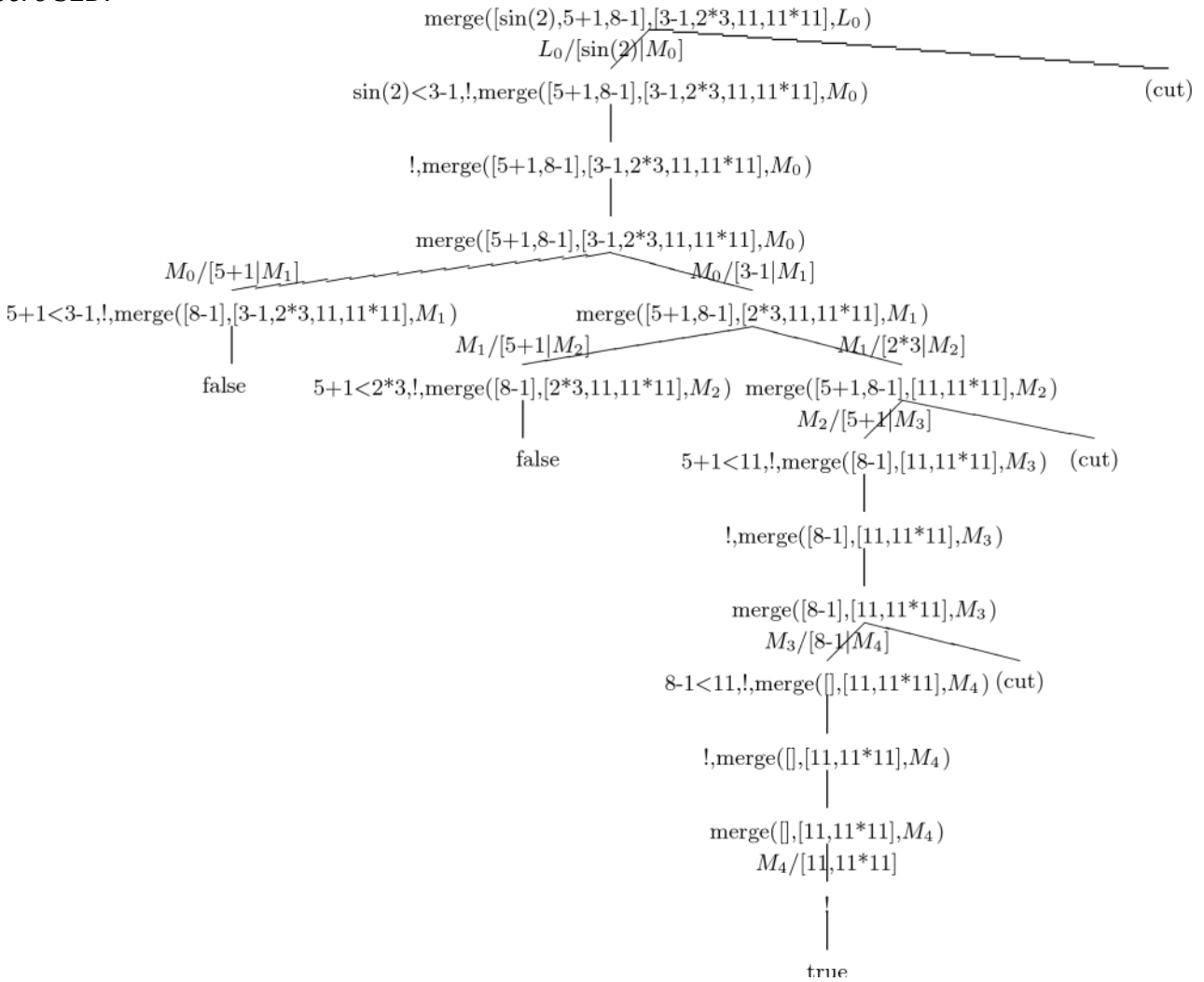
alfa-beta:



I nodi non colorati sono quelli che vengono tagliati nell'algoritmo alfa-beta.

Esercizio 3

Albero SLD:



Esercizio 4

```

vect([X|_], 1, X) :- !.
vect([_|T], I, E1) :- I1 is I-1, vect(T, I1, E1).

```

Esercizio 5

Grafo dei vincoli:



Le variabili sono: A, B, C, D, E, F ed il loro dominio è: {rosso, verde, blu}.

I vincoli sono di tipo binario e sono del tipo $A \neq B$ per ogni arco indicato nel grafo precedente.

Si deve:

- Assegnare prima la variabile con meno valori residui (Minimum Remaining Value - MRV),
- A parità dei domini assegnare prima la variabile più vincolante, coinvolta in più vincoli con le altre variabili (euristica del grado),
- In caso di parità, scegliere la variabile in ordine alfabetico ($A < B < C < D < E < F$),
- Assegnare i valori nell'ordine in cui compaiono nel dominio (rosso, verde, blu),
- Usare forward checking per propagare i vincoli

1) A : {rosso, verde, blu} B : {rosso, verde, blu} C : {rosso, verde, blu} D : {rosso, verde, blu} E : {rosso, verde, blu} F : {rosso, verde, blu}	2) A : {verde, blu} B = rosso C : {rosso, verde, blu} D : {verde, blu} E : {verde, blu} F : {verde, blu}	3) A : {blu} B = rosso C : {rosso, blu} D : {verde, blu} E = verde F : {blu}	4) A : {blu} B = rosso C : {rosso} D : {verde} E = verde F = blu
5) A = blu B = rosso C : {rosso} D : {verde} E = verde F = blu	6) A = blu B = rosso C = rosso D : {verde} E = verde F = blu	7) A = blu B = rosso C = rosso D = verde E = verde F = blu	

Al primo passo si sceglie B perché tutte le variabili hanno dominio pieno e tra quelle coinvolte in più vincoli (4: B, E, F) è la prima in ordine alfabetico. Il valore assegnato è il primo tra quelli ammissibili, cioè rosso.

Al secondo passo A, D, E, F sono le variabili con meno valori residui; tra queste E ed F sono quelle coinvolte in più vincoli (A è coinvolta in 3 vincoli e D in 2) dunque, in base all'ordine alfabetico, E viene scelta per l'assegnamento. In questo caso, il valore da assegnare è verde.

Al passo successivo, A ed F sono le variabili con meno valori residui ed F è coinvolta in 4 vincoli rispetto ai 3 di A, dunque si assegna ad F l'unico valore ammissibile, cioè blu.

Ora A, C e D hanno il minor numero di valori residui, A e C sono coinvolti in 3 vincoli rispetto ai 2 di D, dunque la scelta della variabile da assegnare è A e il valore che assume (l'unico possibile) è blu.

Tra C e D, che hanno ancora 1 valore ammissibile nel dominio, si sceglie C perché è coinvolta in un maggior numero di vincoli di D. Il valore che assume è ovviamente rosso.

Infine si assegna il valore verde all'unica variabile rimasta, D.

Esercizio 6

Vedi slides del corso.

FONDAMENTI DI INTELLIGENZA ARTIFICIALE – 2° parte

21 Dicembre 2011 – Soluzioni

Esercizio 7

Vedi slides del corso.

Esercizio 8

Vedi slides del corso.

Esercizio 9

```
%% solve(GOAL)
%% Meta-interprete per clausole Prolog estese con l'operatore OR ";"
solve(true).
solve((A, B)) :- !, solve(A), solve(B).
solve((A; B)) :- solve(A).
solve((A; B)) :- solve(B), !.
solve(Head) :- clause(Head, Body), solve(Body).
```