

**COMPITO DI INTELLIGENZA ARTIFICIALE Parte I
e FONDAMENTI DI INTELLIGENZA ARTIFICIALE**

8 Luglio 2003

Durata: 2 h (totale su 32 punti)

Esercizio 1 (punti 6)

Si scriva un programma Prolog `conclist (Lin1, Lin2, Lout3)` che date due liste di elementi `Lin1` e `Lin2`, produca una lista in uscita `Lout` contenente come elementi quelli di `Lin1` e `Lin2` intervallati. Poiché `Lin1` e `Lin2` non hanno necessariamente la stessa lunghezza, terminate le coppie, gli elementi rimanenti (o in `Lin1` o in `Lin2`), vengono inseriti inalterati in `Lout3`.

Esempio:

```
?-conclist([a, b, c], [3,7,9,0], X).
```

restituisce

```
X= [a, 3, b, 7, c, 9, 0]
```

Si scrivano esplicitamente tutti i predicati Prolog usati nella soluzione.

Esercizio 2 (punti 8)

Il seguente programma Prolog:

```
sumtree(n(TL,TR),N) :-  
    sumtree(TL,L),  
    sumtree(TR,R),  
    N is L+R.  
sumtree(X,X).
```

dà errore in backtracking. Si inserisca un cut che permetta di avere un comportamento corretto e si rappresenti l'albero SLD corrispondente alla invocazione

```
:- sumtree(n(1,n(2,3)),N).
```

Facoltativo:

Si mostri come si poteva risolvere il problema senza usare il cut (usando la negazione).

Esercizio 3 (punti 8)

Si rappresentino le frasi seguenti in logica dei predicati del I ordine:

- Chiunque è in vacanza e ha tanti soldi è felice.
- Chiunque vince al superenalotto ha tanti soldi.
- Chiunque gioca a qualche gioco ed è fortunato vince a quel gioco.
- Claudio è in vacanza.
- Claudio è fortunato.
- Claudio gioca al superenalotto.

Si traducano in forma a clausole e si dimostri, tramite la risoluzione, che esiste qualcuno che è felice.

Esercizio 4 (punti 6)

Si considerino i seguenti vincoli:

$X :: [1..10]$, $Y :: [1..25]$, $K :: [3..15]$, $Z :: [2..19]$,
 $X > Y$, $Y > K$, $X \geq Z$, $Y \leq Z$, $Z \neq K$.

Si calcolino i domini delle variabili dopo avere applicato l'arc-consistenza al problema iniziale. A partire dal problema così ridotto, si mostri l'albero di ricerca fino alla prima soluzione relativo alla strategia forward checking.

Esercizio 5 (punti 4)

Si spieghino brevemente le strategie non informate sottolineandone vantaggi e svantaggi, si schematizzi poi un algoritmo generale per il loro trattamento.

SOLUZIONE

Esercizio 1

```
conclist([],A,A).
conclist(A,[],A).
conclist([E1|Rest],[E2|Rest1],[E1,E2|Rest3]):-
    conclist(Rest,Rest1,Rest3).
```

Esercizio 2

È necessario inserire il cut per evitare che la seconda clausola venga invocata quando il termine al primo argomento è del tipo $n/2$. Per questo si può inserire il cut come segue:

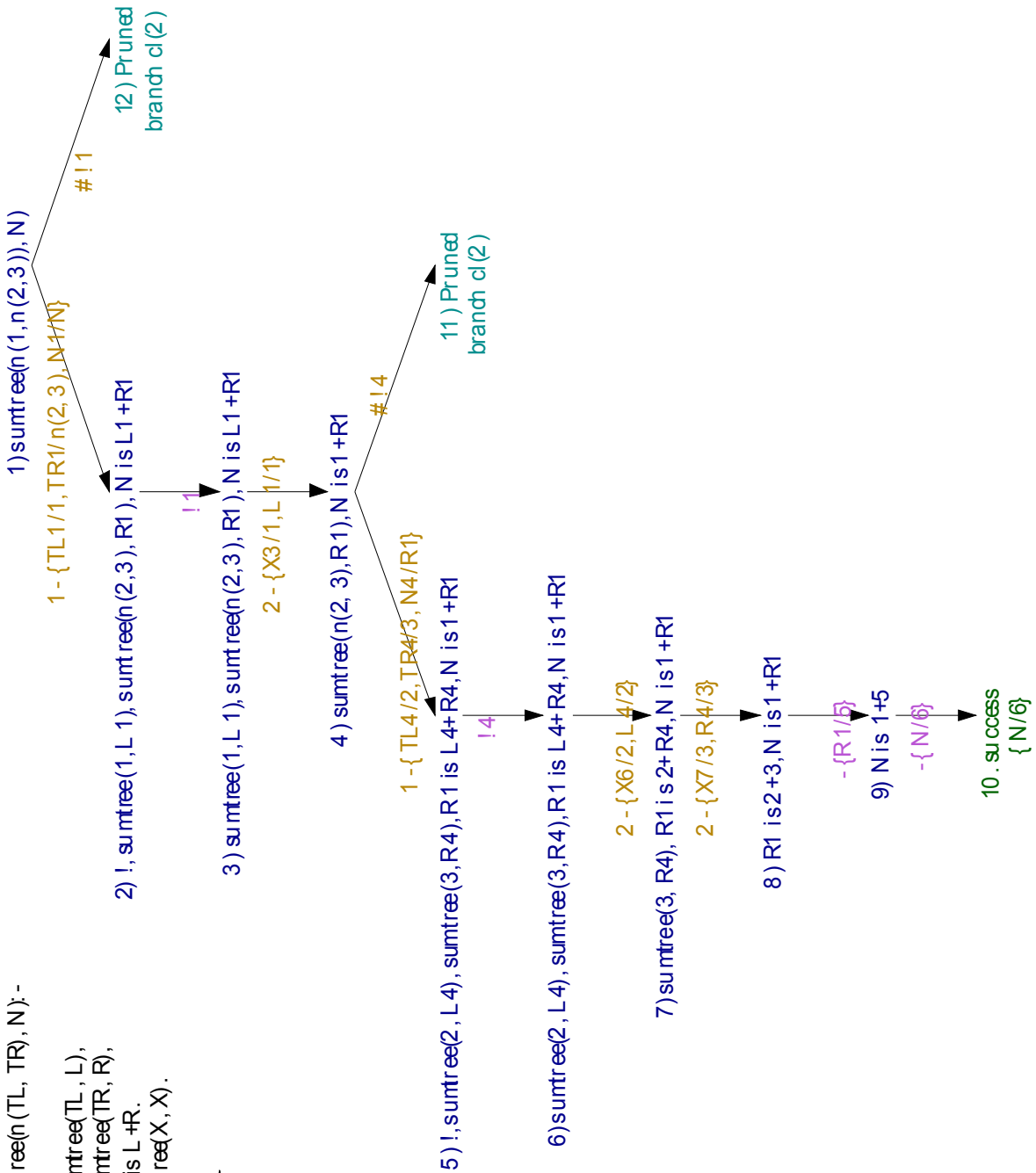
```
sumtree(n(TL,TR),N) :- !,
    sumtree(TL,L),
    sumtree(TR,R),
    N is L+R.
sumtree(X,X).
```

Per ottenere lo stesso comportamento si poteva fare affidamento alla negazione; in questo caso il secondo ramo deve contenere la condizione opposta a quella della prima clausola (prima del cut stesso). In questo caso, l'unica condizione è l'unificazione, quindi:

```
sumtree(n(TL,TR),N) :-
    sumtree(TL,L),
    sumtree(TR,R),
    N is L+R.
sumtree(X,X) :- not( X=n(_,_) ).
```

L'albero SLD (nel caso del cut) è il seguente:

1 sumtree(n(TL, TR), N):-
 !,
 sumtree(TL, L),
 sumtree(TR, R),
 N is L+R.
 2 sumtree(X, X).



Esercizio 3

$\forall X \text{ vacanza}(X) \text{ and } \text{ha_soldi}(X) \Rightarrow \text{felice}(X)$
 $\forall X \text{ vince}(X, \text{superenalotto}) \Rightarrow \text{ha_soldi}(X)$
 $\forall X \forall Y \text{ gioca}(X, Y) \text{ and } \text{fortunato}(X) \Rightarrow \text{vince}(X, Y)$
 $\text{vacanza}(\text{claudio}).$
 $\text{fortunato}(\text{claudio}).$
 $\text{gioca}(\text{claudio}, \text{superenalotto})$

Goal: $\exists Y$ felice(Y).

Forma a clausole:

C1: not vacanza(X) or not ha_soldi(X) or felice(X).

C2: not vince(X,superen) or ha_soldi(X)

C3: not gioca(X,Y) or not fortunato(X) or vince(X,Y)

C4: vacanza(claudio).

C5: fortunato(claudio).

C6: gioca(claudio,superenalotto)

Goal: not felice(Y).

C7=Goal e C1: not vacanza(Y) or not ha_soldi(Y)

C8= C7 e C4: not ha_soldi(claudio)

C9=C8 e C2: not vince(claudio,superen)

C10=C9 e C3: not gioca(claudio,superen) or not fortunato(claudio)

C11=C10 e C5: not gioca(claudio,superen)

C12=C11 e C6: clausola vuota

Esercizio 4

$X :: [1..10]$, $Y :: [1..25]$, $K :: [3..15]$, $Z :: [2..19]$,
 $X > Y$, $Y > K$, $X \geq Z$, $Y \leq Z$, $Z \neq K$.

Situazione della rete dopo avere applicato l'arc-consistenza

$X :: [5..10]$

$Y :: [4..9]$

$Z :: [4..10]$

$K :: [3..8]$

Albero di ricerca fino alla prima soluzione relativo alla strategia forward checking.

