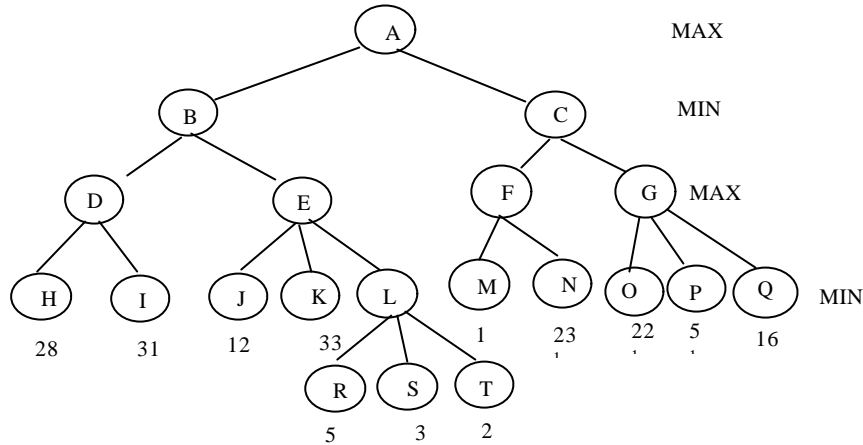


**COMPITO DI INTELLIGENZA ARTIFICIALE 1 (e FONDAMENTI DI IA)**

**18 Marzo 2003 – tempo a disposizione 2 ore**

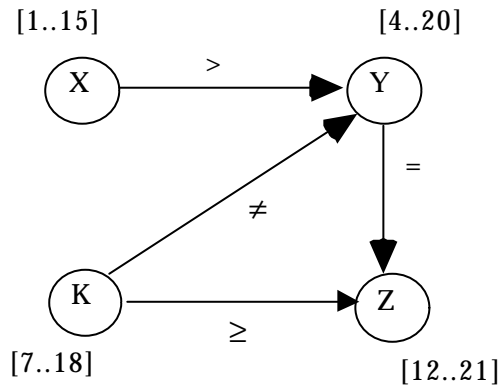
**Esercizio 1 (punti 5)**

Dato il seguente albero di ricerca per un gioco a due giocatori, si mostri quale mossa selezionerà MAX, secondo l'algoritmo MIN-MAX. Si mostri inoltre come può essere ridotto l'albero applicando i tagli alfa-beta.



**Esercizio 2 (punti 7)**

Si consideri la seguente rete di vincoli:



$X::[1..15]$ ,  $Y::[4..20]$ ,  $K::[7..18]$ ,  $Z::[12..21]$ ,  
 $X > Y$ ,  $Y = Z$ ,  $K \geq Z$ ,  $K \neq Y$

Si applichi l'arc-consistenza alla rete iniziale, mostrando i domini finali delle variabili. Si trovi poi una soluzione usando il Forward Checking con euristica di selezione della variabile first-fail.

### Esercizio 3 (punti 7)

Si traducano le seguenti frasi nella logica dei predicati del primo ordine, poi in forma a clausole:

- 1) Tutte le bimbe sognano di essere principesse
- 2) Chi sogna di essere una principessa, lo diventa
- 3) Tutte le bimbe hanno un papà e ciascuna vuole sposare il proprio
- 4) Chiunque una principessa voglia sposare è a sua volta un principe
- 5) Elena è una bimba

Si usi poi il principio di risoluzione per dimostrare che il padre di Elena è un principe.

### Esercizio 4 (punti 7)

Dato il seguente programma Prolog:

```
min_dom(M,L) :-  
    member(B,L), B<M,!.  
  
member(X,[X|_]).  
member(X,[_|_]) :- member(X,_) .
```

si scriva cosa fa il predicato `min_dom/2` in generale e si rappresenti l'albero SLDNF relativa alla seguente invocazione:

```
?- not min_dom(2,[4,2]).
```

### Esercizio 5 (punti 6)

Si scriva un programma Prolog `elimina(EL,L,LOUT)` che data una lista di elementi `L`, elimini da `L` gli elementi che sono uguali (predicato `==`) ad `EL` restituendo il risultato in `LOUT`.

Esempio:

```
?- elimina(p(1,1),[p(1,B),p(1,1),p(1,3),p(1,1)],Lout).
```

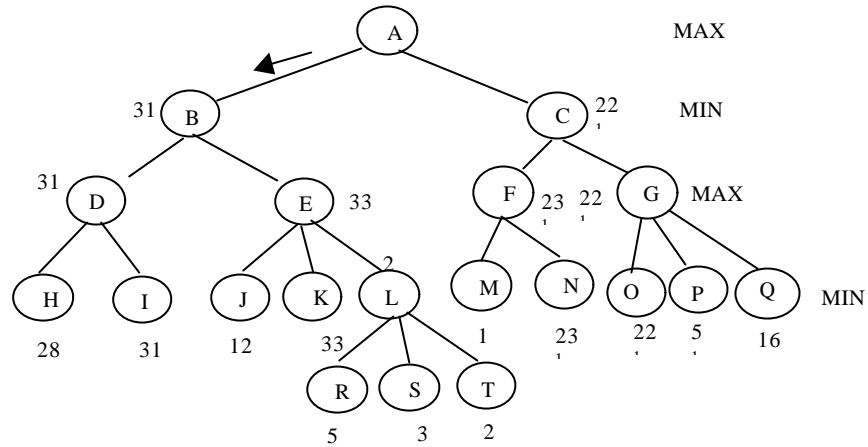
Restituisce:

```
Lout = [p(1,B),p(1,3)]
```

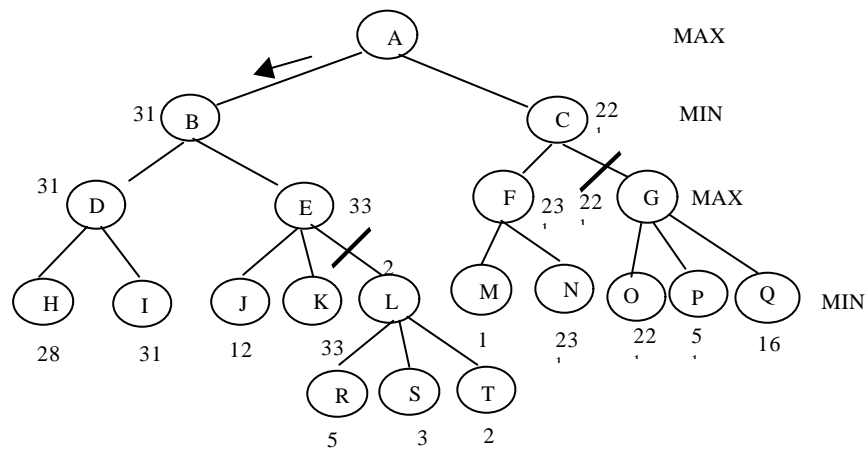
# SOLUZIONE

## Esercizio 1

### MIN MAX



### TAGLI ALFA BETA



## Esercizio 2

Arc consistenza alla rete iniziale

$X::[13..15]$ ,  $Y::[12..14]$ ,  $K::[12..18]$ ,  $Z::[12..14]$

Ho 3 variabili con dominio contenente 3 elementi, quindi ne scelgo una a caso

Istanzio X

$X = 13$

$Y::[12]$ ,  $K::[12..18]$ ,  $Z::[12..14]$

Istanzio Y

$X=13$ ,  $Y=12$ ,

$K :: [13..18], Z :: [12]$

Istanzio Z

$X=13, Y=12, Z=12,$

$K :: [13..18]$

Istanzio K e ottengo la prima soluzione

$X=13, Y=12, K=13, Z=12$

### Esercizio 3

Logica:

- 1  $\forall X, \text{bimba}(X) \Rightarrow \text{sogna}(X, \text{principessa})$
- 2  $\forall X, \text{sogna}(X, \text{principessa}) \Rightarrow \text{principessa}(X)$
- 3  $\forall X, \text{bimba}(X) \Rightarrow \exists Y \text{ papa}(X, Y) \wedge \text{sposa}(X, Y)$
- 4  $\forall X \forall Y, \text{principessa}(X), \text{sposa}(X, Y) \Rightarrow \text{principe}(Y)$
- 5  $\text{bimba}(\text{elena})$

Query:  $\exists Y \text{ papa}(\text{elena}, Y) \wedge \text{principe}(Y)$

Query negata:

**G**  $\sim \text{papa}(\text{elena}, Y) \vee \sim \text{principe}(Y)$

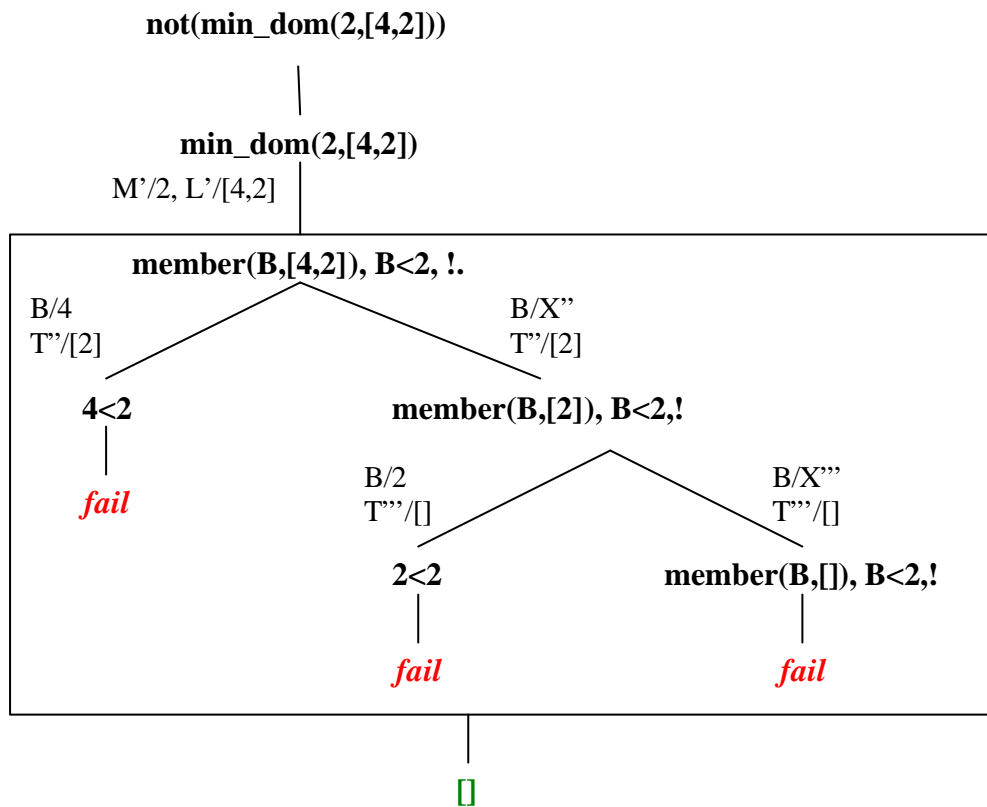
Forma a clausole:

- 1  $\sim \text{bimba}(X) \vee \text{sogna}(X, \text{principessa})$
- 2  $\sim \text{sogna}(X, \text{principessa}) \vee \text{principessa}(X)$
- 3'  $\sim \text{bimba}(X) \vee \text{papa}(X, \text{padre}(X))$
- 3''  $\sim \text{bimba}(X) \vee \text{sposa}(X, \text{padre}(X))$
- 4  $\sim \text{principessa}(X) \vee \sim \text{sposa}(X, Y) \vee \text{principe}(Y)$
- 5  $\text{bimba}(\text{elena})$

Risoluzione:

- 6 (G+3')  $\sim \text{principe}(\text{padre}(\text{elena}))$
- 7 (6+4)  $\sim \text{principessa}(\text{elena}) \vee \sim \text{sposa}(\text{elena}, \text{padre}(\text{elena}))$
- 8 (7+2)  $\sim \text{sogna}(\text{elena}, \text{principessa}) \vee \sim \text{sposa}(\text{elena}, \text{padre}(\text{elena}))$
- 9 (8+3'')  $\sim \text{sogna}(\text{elena}, \text{principessa}) \vee \sim \text{bimba}(\text{elena})$
- 10 (9+5)  $\sim \text{sogna}(\text{elena}, \text{principessa})$
- 11 (10+1)  $\sim \text{bimba}(\text{elena})$
- 12 (11+5) clauola vuota

### Esercizio 4



### Esercizio 5

```

elimina(_, [], []).
elimina(X, [Y|T], L) :- X==Y, !, elimina(X, T, L).
elimina(X, [Y|T], [Y|L]) :- elimina(X, T, L)

```