

## FONDAMENTI DI INTELLIGENZA ARTIFICIALE LS – PRIMA PARTE

15 Luglio 2010 – Tempo a disposizione 2h – Risultato 32/32 punti

### Esercizio 1 (punti 6)

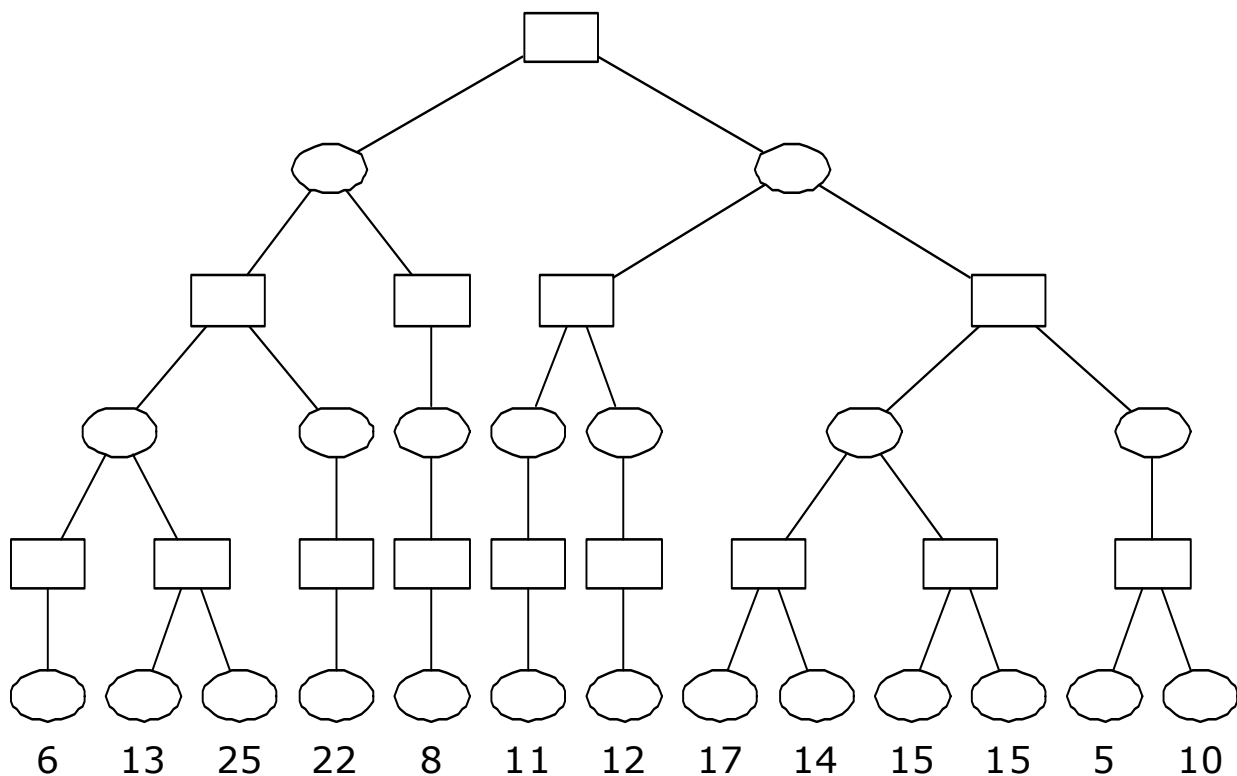
Si formalizzino il logica dei predicati del I ordine le seguenti frasi:

- *Ogni bimbo ama ogni caramella*
- *Chiunque ami una caramella non è un nutrizionista*
- *Chiunque mangi qualche zucca è un nutrizionista*
- *Chiunque acquisti qualsiasi zucca o la intaglia o la mangia (or esclusivo)*
- *Giovanni acquista una zucca*
- *Giovanni è un bimbo*
- *Golia è una caramella*

Dimostrare poi applicando il principio di risoluzione che: *Giovanni intaglia una zucca*

### Esercizio 2 (punti 5)

Si consideri il seguente albero di gioco in cui la valutazione dei nodi terminali è dal punto di vista del primo giocatore. Si assuma che il primo giocatore sia Max.



Si mostri come gli algoritmi min-max e alfa-beta risolvono il problema

### Esercizio 3 (punti 7)

Il seguente programma Prolog verifica se in una lista tutti gli elementi sono diversi:

```
alldiffer(L) :- not(twoequal(L)).
twoequal(L) :- membrest(X,T,L), membchk(X,T), !.
membrest(X,T,[X|_]).
membrest(X,T,[_|L]) :- membrest(X,T,L).
```

Si mostri l'albero di derivazione SLDNF relativo al goal

```
:- alldiffer([p(r,a),p(A,r),p(a,R)]).
```

Si noti che il predicato membchk/2 è da intendersi come il predicato member/2 usuale.

#### Esercizio 4 (punti 6)

Nel seguente problema si tratta di trovare una sistemazione delle cifre da 1 a 8 nella scacchiera raffigurata sotto, in modo tale che ogni cifra non sia consecutiva a nessuna delle cifre in caselle adiacenti (in orizzontale, verticale o diagonale). Ad esempio la configurazione in figura non sarebbe una soluzione, perché 1 e 2 sono in caselle adiacenti, come anche 2 e 3, 7 e 8, 5 e 6.

Si formuli il problema come un problema di soddisfacimento di vincoli e lo si risolva tramite il *forward checking*.

	1	
2	7	4
3	8	6
	5	

Nel mostrare come si ottiene la soluzione, si scelga, ad ogni passo, come variabile da istanziare quella che ha il valore più piccolo nel dominio (e in caso di parità, si istanzi la variabile soggetta a più vincoli. Se ancora si presentano casi di parità in base, si selezioni prima la variabile con indice più basso.

Come euristica di selezione del valore, si assegnino i valori in ordine crescente.

#### Esercizio 5 (punti 5)

Dato un insieme di studenti e i voti ottenuti negli esami, rappresentati come fatti del tipo: `studente(Nome, Voto)`.

si definisca il predicato `studMedia(Nome, Media)` che dato il nome di uno studente (Nome) ne determina la media in Media.

#### Esercizio 6 (punti 3)

Si dia la definizione di *euristica ammissibile* e *euristica monotona*. La monotonicità della  $f$  euristica ne garantisce l'ammissibilità? Perché sono importanti le euristiche ammissibili? E quelle monotone?

## SOLUZIONE

### Esercizio 1

1.  $\forall X \forall Y ( \text{ bimbo}(X) \text{ and } \text{ caramella}(Y) \rightarrow \text{ ama}(X,Y) )$
2.  $\forall X \forall Y ( \text{ caramella}(Y) \text{ and } \text{ ama}(X,Y) \rightarrow \neg \text{ nutriz}(X) )$
3.  $\forall X ((\exists Y ( \text{ zucca}(Y) \text{ and } \text{ mangia}(X,Y))) \rightarrow \text{ nutriz}(X))$
4.  $\forall X \forall Y ( \text{ zucca}(Y) \text{ and } \text{ compra}(X,Y) \rightarrow \text{ intaglia}(X,Y) \text{ xor } \text{ mangia}(X,Y) )$
5.  $\exists Y ( \text{ zucca}(Y) \text{ and } \text{ compra}(\text{giovanni},Y))$
6.  $\text{ bimbo}(\text{giovanni})$
7.  $\text{ caramella}(\text{golia})$
8.  $\neg (\exists Y ( \text{ zucca}(Y) \text{ and } \text{ intaglia}(\text{giovanni},Y)) )$   
equivale a :  $\forall Y ( \text{ zucca}(Y) \text{ and } \text{ intaglia}(\text{giovanni},Y) )$

### Clausole:

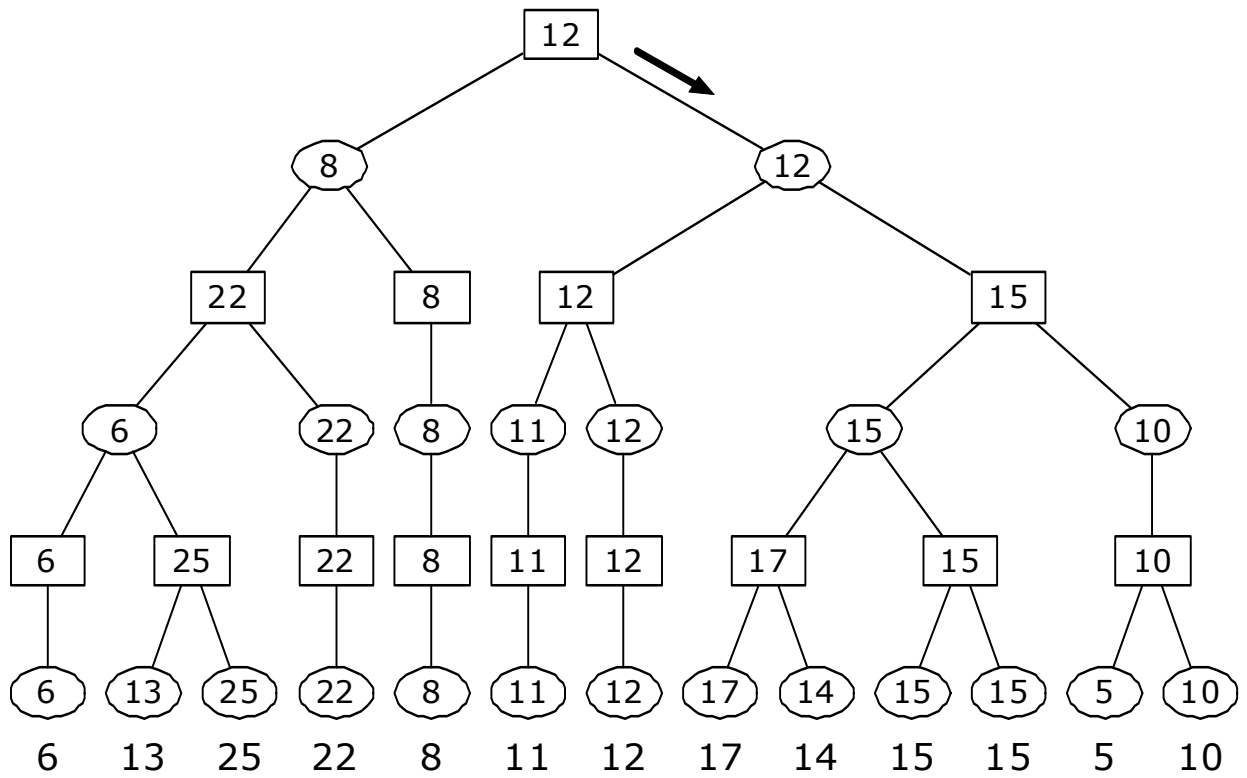
- 1  $\neg \text{ bimbo}(X) \text{ or } \neg \text{ caramella}(Y) \text{ or } \text{ ama}(X,Y)$
- 2  $\neg \text{ caramella}(Y) \text{ or } \neg \text{ ama}(X,Y) \text{ or } \neg \text{ nutriz}(X)$
- 3  $\neg \text{ zucca}(Y) \text{ or } \neg \text{ mangia}(X,Y) \text{ or } \text{ nutriz}(X)$
- 4  $\neg \text{ zucca}(Y) \text{ or } \neg \text{ compra}(X,Y) \text{ or } \text{ intaglia}(X,Y) \text{ or } \text{ mangia}(X,Y)$
- 5  $\neg \text{ zucca}(Y) \text{ or } \neg \text{ compra}(X,Y) \text{ or } \neg \text{ intaglia}(X,Y) \text{ or } \neg \text{ mangia}(X,Y)$
- 6  $\text{ zucca}(c)$
- 7  $\text{ compra}(\text{giovanni},c)$
- 8  $\text{ bimbo}(\text{giovanni})$
- 9  $\text{ caramella}(\text{golia})$
- 10  $\neg \text{ zucca}(Y) \text{ or } \neg \text{ intaglia}(\text{giovanni},Y)$

### Risoluzione:

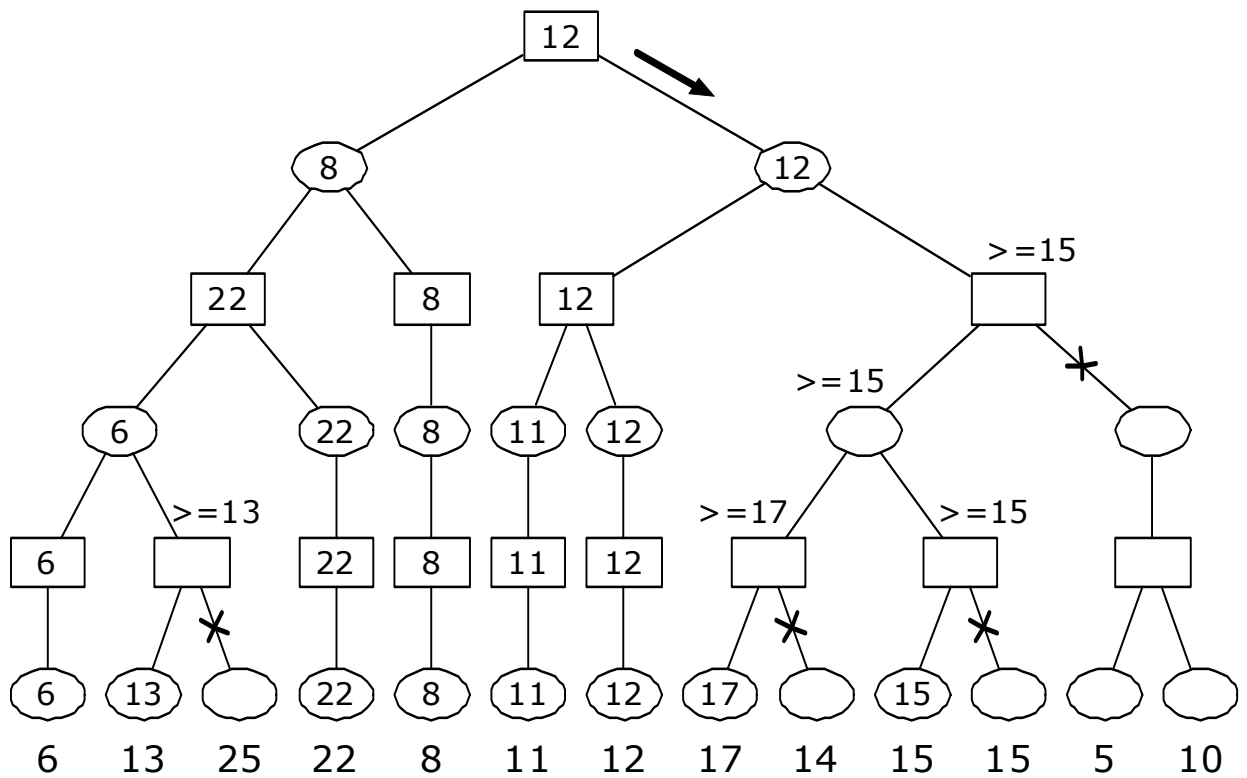
11.  $\text{ ama}(\text{giovanni},\text{golia})$  da 8+9+1
12.  $\neg \text{ nutriz}(\text{giovanni})$  da 11+9+2
13.  $\text{ intaglia}(\text{giovanni},c) \text{ or } \text{ mangia}(\text{giovanni},c)$  da 6+7+4
14.  $\text{ mangia}(\text{giovanni},c)$  da 6+10+13
15.  $\text{ nutriz}(\text{giovanni})$  da 3+13+6
16. vuota da 15+12

## Esercizio 2

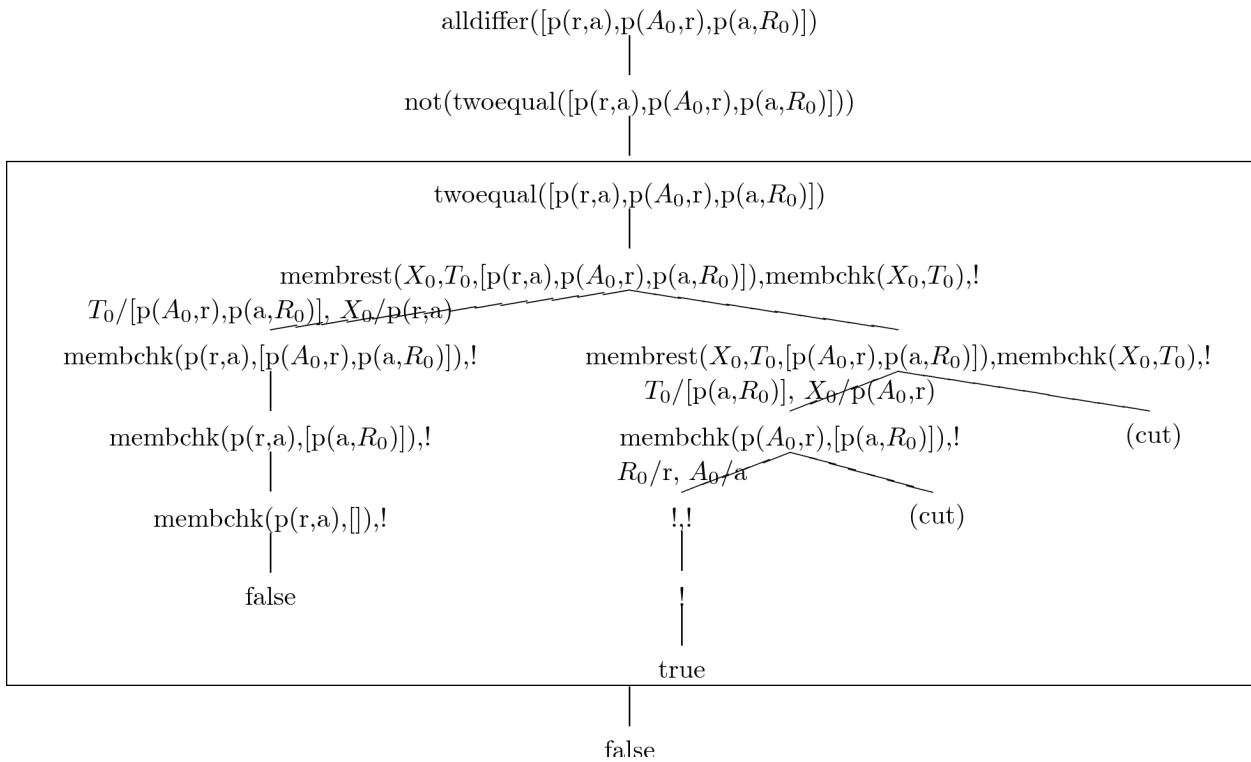
min-max:



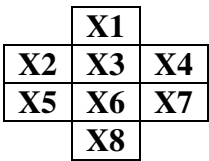
Alfa-beta:



### Esercizio 3



### Esercizio 4



Formulazione CSP:

- *variabili*: X1, ..., X8, associate alle caselle come in figura
- *domini*: valori da 1 a 8
- *vincoli*:
  - $\forall i,j, i \neq j \rightarrow X_i \neq X_j$
  - $\text{adiacente}(i,j) \rightarrow X_i \neq X_{j+1}, X_i \neq X_{j-1}$
  - $\text{adiacente}(1,2), \text{adiacente}(1,3), \text{adiacente}(1,4), \text{adiacente}(2,3), \text{adiacente}(2,5),$   
 $\text{adiacente}(2,6), \text{adiacente}(3,4), \text{adiacente}(3,5), \text{adiacente}(3,6), \text{adiacente}(3,7),$   
 $\text{adiacente}(4,5), \text{adiacente}(4,7), \text{adiacente}(5,6), \text{adiacente}(5,8), \text{adiacente}(6,7),$   
 $\text{adiacente}(6,8), \text{adiacente}(7,8).$

Forward Checking:

Tutte le variabili hanno inizialmente lo stesso dominio.

Le variabili soggette a più vincoli sono X3 e X6; partiamo da X3.

X3=1

X1	X2	X3	X4	X5	X6	X7	X8
3..8	3..8	<b>1</b>	3..8	3..8	3..8	3..8	2..8

Ora la variabile che ha il valore più piccolo nel dominio è X8 (che ha il valore 2):

X1	X2	X3	X4	X5	X6	X7	X8
3..8	3..8	<b>1</b>	3..8	4..8	4..8	4..8	<b>2</b>

Il valore più piccolo nei domini è 3 ed è nel dominio delle variabili X1, X2, X4. X2 e X4 hanno più vincoli di X1, in quanto X1 ha vincoli solo con X2 e X4, mentre X2 ha vincoli con X5 e X6 e X4 ha vincoli con X6 e X7 (oltre al vincolo di diverso fra tutte le variabili, che è comune a tutte). Istanziamo quindi X2:

X1	X2	X3	X4	X5	X6	X7	X8
5..8	<b>3</b>	<b>1</b>	4..8	5..8	5..8	4..8	<b>2</b>

Ora il valore più piccolo è 4 ed è nei domini di X4 e X7. X4 ha vincoli con 3 variabili (X1, X6 e X7), mentre X7 solo con 2 (X4 e X6). Istanziamo quindi X4:

X1	X2	X3	X4	X5	X6	X7	X8
6..8	<b>3</b>	<b>1</b>	<b>4</b>	5..8	6..8	6..8	<b>2</b>

Il valore più piccolo è nel dominio di X5:

X1	X2	X3	X4	X5	X6	X7	X8
6..8	<b>3</b>	<b>1</b>	<b>4</b>	<b>5</b>	7..8	6..8	<b>2</b>

Il valore più piccolo è nel dominio di X1 e X7; X7 ha un vincolo con X6 che X1 non ha:

X1	X2	X3	X4	X5	X6	X7	X8
7..8	<b>3</b>	<b>1</b>	<b>4</b>	<b>5</b>	8	<b>6</b>	<b>2</b>

X1 ha l'elemento più piccolo nel dominio e viene istanziato a 7. Infine si istanzia X6 a 8.

La soluzione è:

	<b>7</b>	
<b>3</b>	<b>1</b>	<b>4</b>
<b>5</b>	<b>8</b>	<b>6</b>
	<b>2</b>	

### Esercizio 5

```
studMedia(Nome,Media):- findall(Voto,studente(Nome,Voto),List),
                        media(List, Media).
```

```
%semplice predicato per calcolare la media su una lista di numeri
media(Lista,Media) :- length(Lista,LunghezzaLista),
                    sum(Lista,Somma),
                    Media is Somma / LunghezzaLista.
```

```
sum([],0).
```

```
sum([A|B],N):- sum(B,N1),N is N1+1.
```