FONDAMENTI DI INTELLIGENZA ARTIFICIALE (6 CFU)

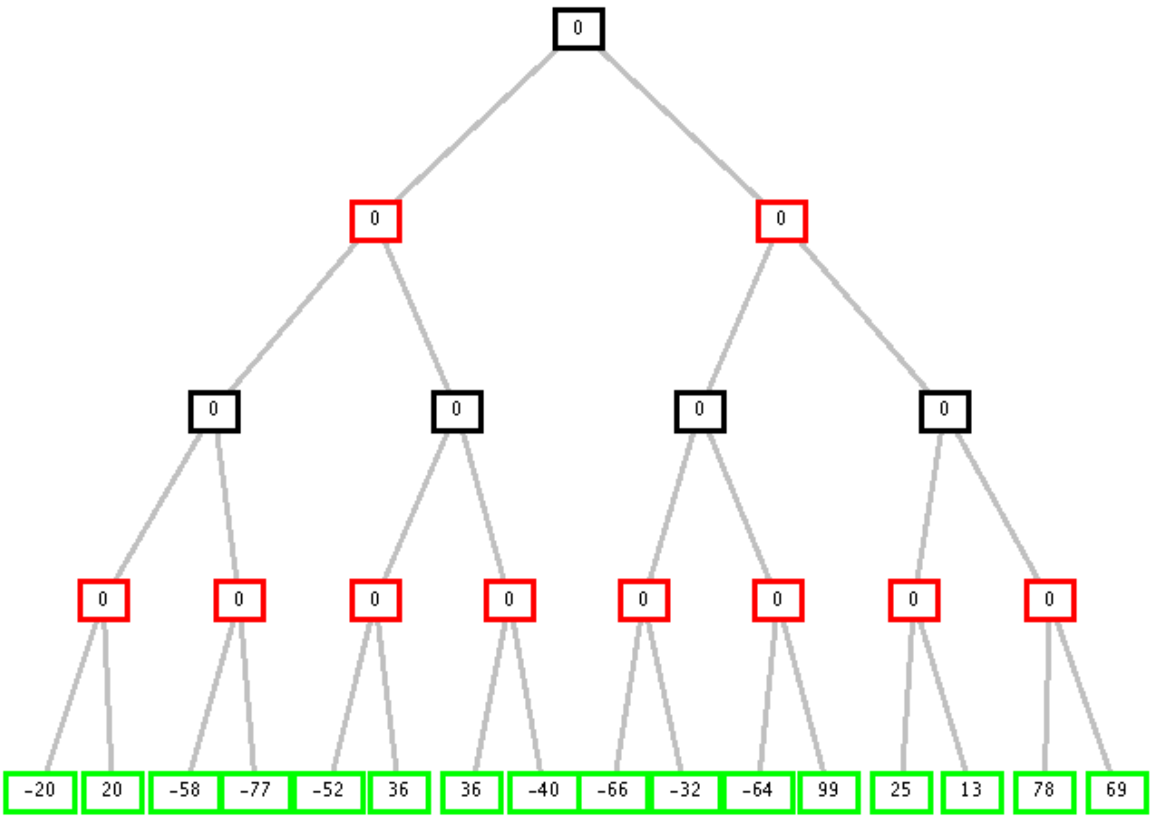
13 Gennaio 2015 – Tempo a disposizione: 2 h – Risultato: 32/32 punti

Esercizio 1 (6 punti)
Si esprimano in logica dei predicati del I ordine le seguenti frasi:

- A tutti i bambini piacciono tutte le torte
- Se a un bambino piace qualcosa, allora lo mangia (e viceversa: se mangia qualcosa, allora gli piace).
- La "honorè" è una torta
- Federico non mangia la "honorè".

Si usi poi il principio di risoluzione per dimostrare che **Federico non è un bambino**. Si usi il seguente vocabolario: predicati: **b(X)** X è un bambino, **t(Y)** Y è una torta, **piace(X,Y)** a X piace Y, **mangia(X,Y)** X mangia Y; costanti: **fed** (federico), **hon** (honorè).

Esercizio 2 (6 punti)
Si consideri il seguente albero di gioco in cui la valutazione dei nodi terminali è dal punto di vista del primo giocatore (*MAX*). Si mostri come gli algoritmi *min-max* e *alfa-beta* risolvono il problema.



Esercizio 3 (5 punti)
Definire nel linguaggio *Prolog* il predicato **`extract(X,L,Lout)`**, che dato un termine **`X`** ground e una lista di termini ground **`L`**, ha successo con **`Lout`** lista che contiene tutte le occorrenze di **`X`** in **`L`**. Esempi:

```
?- extract(1, [1,4,5,4,1,1], L).
yes    L=[1,1,1]
?- extract (2, [1,4,5,4,1,1], L).
yes L=[]
?- extract (f(5), [f(1),4,f(5),4,1,1], []).
no
```

Esercizio 4 (9 punti)

Si consideri il problema dell'ordinamento in senso crescente di una lista di n elementi (x1, ... , xn). Si assuma di poter determinare l'ordinamento di una qualsiasi coppia di elementi della lista, e che l'unico tipo di azioni eseguibili sia lo scambio di una coppia di elementi. Nella formulazione del problema come un problema di ricerca, uno stato corrisponde a una lista composta da una data permutazione degli n elementi della lista da ordinare. Lo stato iniziale corrisponde alla lista da ordinare, lo stato obiettivo corrisponde alla lista ordinata contenente gli stessi elementi. Si considerino come operatori tutti i possibili scambi di coppie di elementi non ordinati della lista corrispondente a uno stato. Lo spazio degli stati è quindi costituito da un grafo orientato tale che i successori di ogni nodo x corrispondono a tutte le liste ottenibili da quella del nodo x scambiando coppie di elementi non ordinati.

a) Si supponga che lo scambio di due elementi della lista abbia costo costante, pari all'unità. Si consideri poi la seguente euristica: in uno stato in cui vi sono p elementi "fuori posto", l'euristica vale p/2 (arrotondato all'intero superiore). Tale euristica è ammissibile? E' consistente?
b) Si applichi l'algoritmo di ricerca A* (con l'euristica definita nel punto b) alla lista (4 3 2 1), espandendo non più di 5 nodi dell'albero di ricerca (indicare chiaramente l'ordine di espansione dei nodi, e i valori dell'euristica e della funzione di valutazione di ogni nodo generato). Per la generazione dei nuovi stati si applichino gli operatori nel seguente modo: si consideri ogni elemento della lista del nodo da espandere, da sinistra verso destra, e ciascuno degli elementi successivi, ancora da sinistra verso destra. A parità del valore della funzione di valutazione, si espanda il nodo a profondità minore, e a parità di profondità quello generato per primo.

Esercizio 5 (6 punti)

Si descriva brevemente l'idea alla base degli algoritmi di propagazione dei vincoli rispetto alla tecnica "Standard Backtracking", e si delineino sinteticamente le differenze tra "Forward Checking", "Partial Look Ahead" e "Full Look Ahead", mostrandone anche il comportamento sull'esempio seguente:

X0=1
X1,X2,X3::[2,3,4,5]
X0<X1<X2<X3

Fondamenti di Intelligenza Artificiale

13 Gennaio 2015 – Soluzioni

Esercizio 1

Rappresentazione in logica del I ordine:

1. ∀X ∀Y b(X) ∧ t(Y) ⟶ piace(X,Y).
2. ∀X ∀Y b(X) ∧ piace(X,Y) ⟶ mangia(X,Y).
3. ∀X ∀Y b(X) ∧ mangia(X,Y) ⟶ piace(X,Y).
4. t(hon).
5. ┐mangia(fed, hon).

Goal: ┐b(fed).

Trasformazione in clausole:

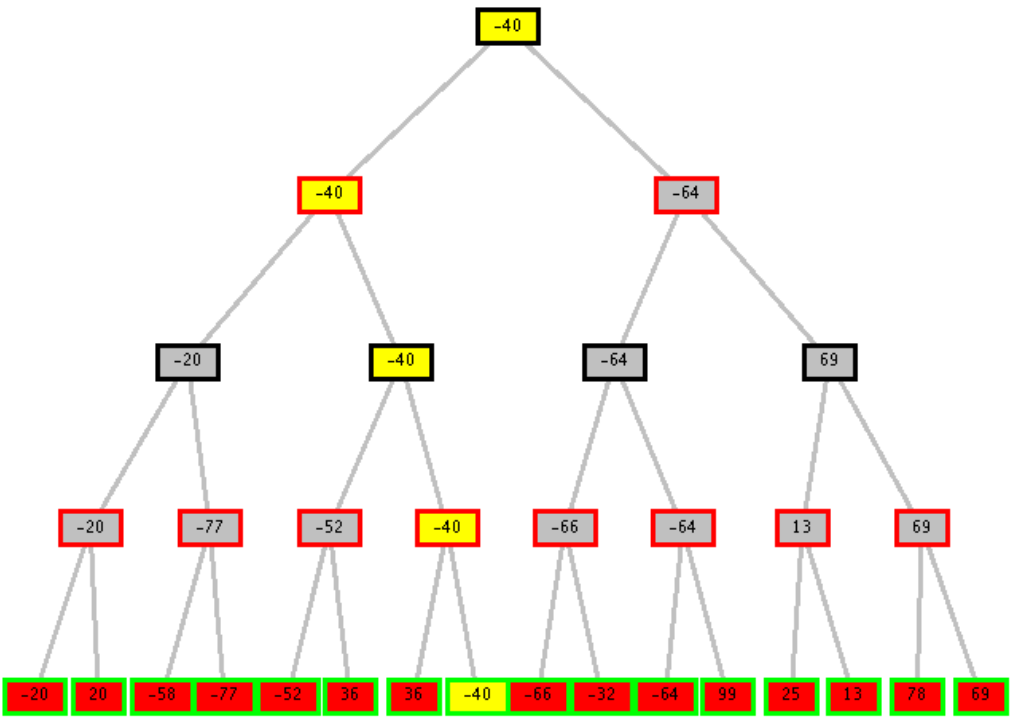
C1: not b(X) or not t(Y) or piace(X,Y).
C2: not b(X) or not piace(X,Y) or mangia(X,Y).
C3: not b(X) or not mangia(X,Y) or piace(X,Y).
C4: t(hon).
C5: not mangia(fed, hon).
GNeg: b(fed).

Risoluzione:

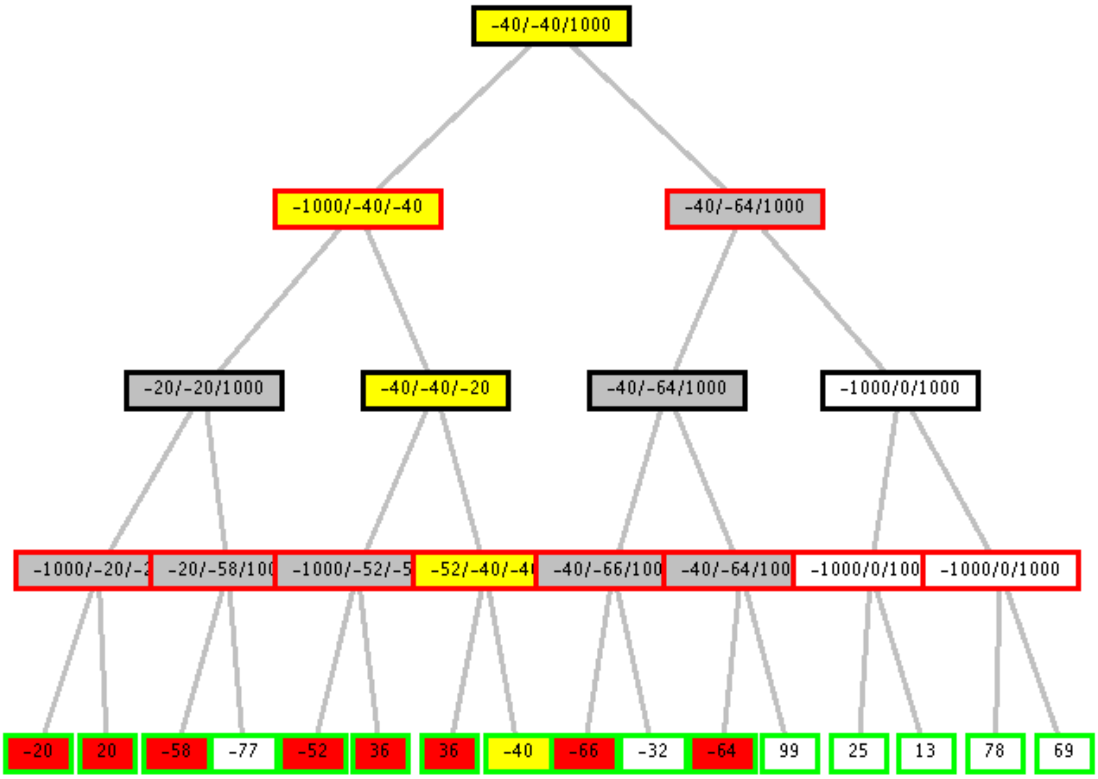
C6: GNeg+C2: not piace(fed,Y) or mangia(fed,Y).
C7: C6 + C5: not piace(fed, hon).
C8: C7 + C1: not b(fed) or not t(hon).
C9: C8 + GNeg: not t(hon).
C10: C9 + C4: clausola vuota.

Esercizio 2

Min-max:



Alfa-Beta:



Esercizio 3

```
extract( _, [], []).
extract( X, [X|T], [X|L]) :- !,extract(X,T,L).
extract( X, [_|T], L) :- extract(X,T,L).
```

Esercizio 4

a) Tenendo conto degli operatori definiti nel punto a), se una lista contiene p elementi fuori posto rispetto allo stato obbiettivo è facile verificare che sarà necessario scambiare non meno di “p/2” coppie di elementi per ordinarla. Quindi la funzione euristica è ammissibile per questo problema. La funzione euristica è anche consistente. A tal scopo osserviamo che, affinchè sia consistente, deve valere:

- h(n) = 0 se n è lo stato obiettivo: vero, poiché l’euristica nello stato obiettivo ha zero nodi fuori posto, e quindi vale zero;
- h(n) <= c(n, a, n’) + h(n’): supponendo che nello stato n vi siano p elementi fuori posto, nel caso migliore saranno necessari p/2 scambi (arrotondato all’intero sup.). Poiché la singola azione consiste in un singolo scambio, nell’ipotesi migliore entrambi gli elementi andranno nella posizione “corretta”, e quindi gli elementi fuori posto (sempre nell’ipotesi migliore) saranno (p-2); l’euristica per il nodo n’ varrà h(n’)=(p-2)/2. Riassumendo, nell’ipotesi del caso migliore, per il generico nodo n varrà sempre:

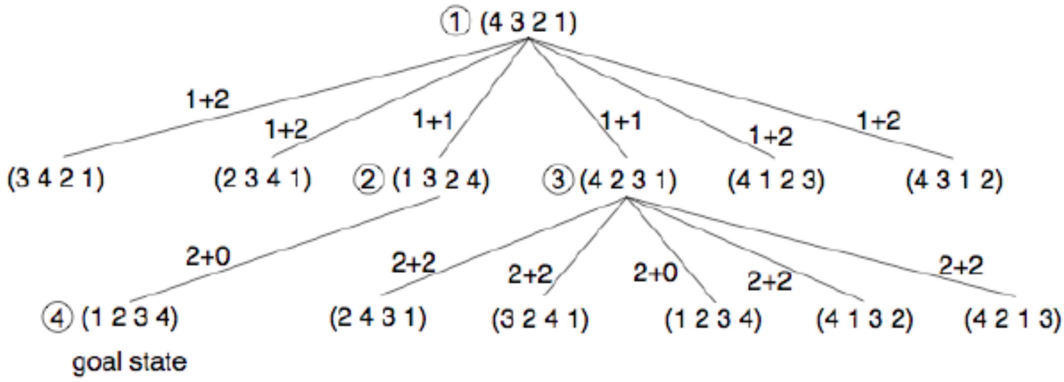
  $$p/2 <= 1 + (p-2)/2$$

  cioè:

  $$p/2 <= 1 + p/2 -1$$

  Da cui si evince che la disuguaglianza è sempre verificata.

c) L'albero di ricerca è mostrato in figura. L'ordine di espansione dei nodi è indicato dai numeri all'interno dei cerchi. Su ogni ramo è indicata la funzione di valutazione del nodo risultante, come somma del pathcost e della funzione euristica.



Esercizio 5

Si vedano le slides del corso per le definizioni.

Sull'esempio, dopo il labeling X1=2, le propagazioni FC, PLA e FLA producono i seguenti domini:

FC	X2,X3::[3,4,5]	
PLA	X2::[3,4]	X3::[3,4,5]
FLA	X2::[3,4]	X3::[4,5]