Fondamenti di Intelligenza Artificiale

25 Gennaio 2018 – Tempo a disposizione: 2 h – Risultato: 32/32 punti

Esercizio 1 (6 punti)

Si formalizzino le seguenti frasi in logica dei predicati:

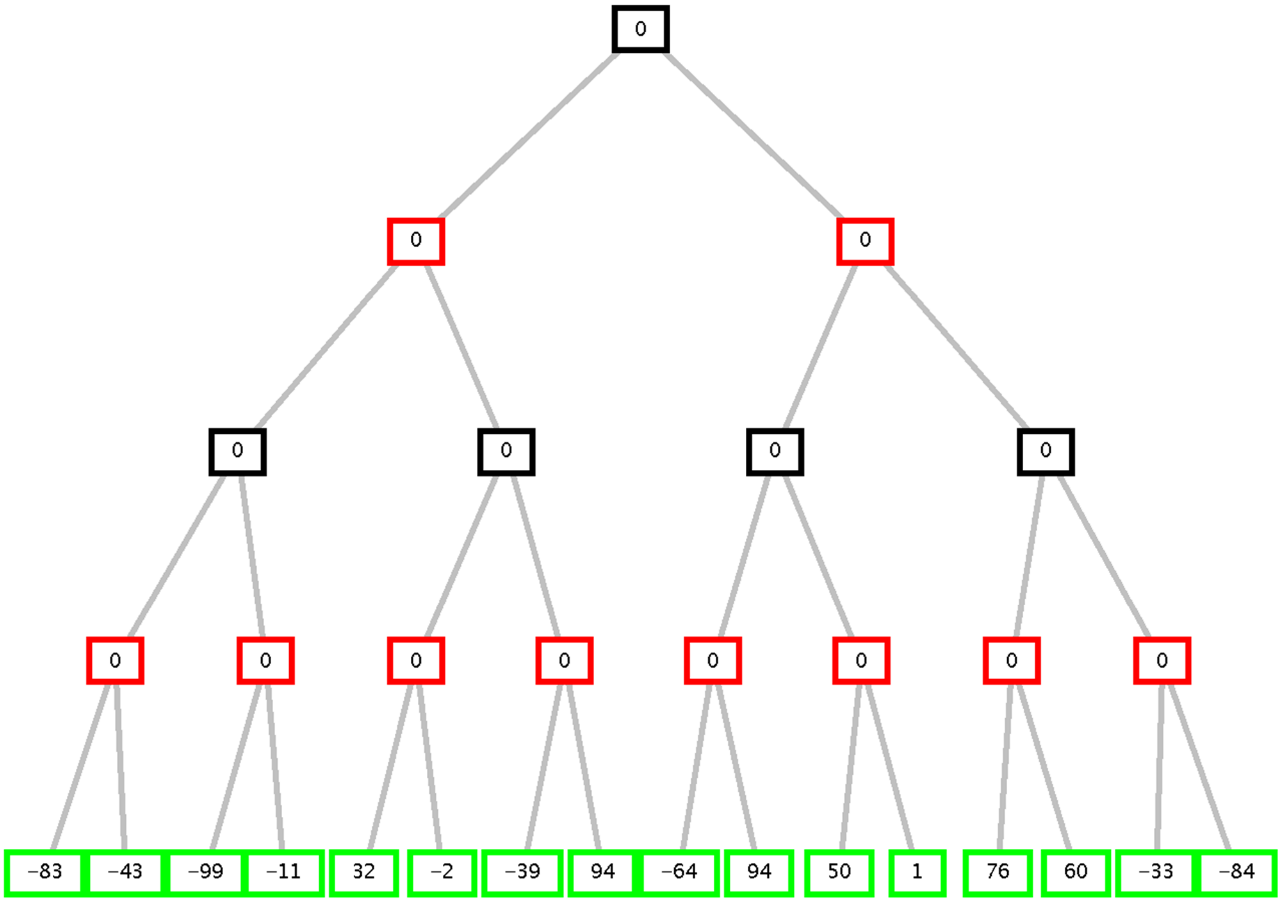
1. Ogni attore invitato al party arriva in ritardo
2. Chiunque arrivi in orario al party, non arriva in ritardo, e viceversa
3. Tutti gli invitati arrivano in orario al party

Le si trasformi in clausole usando i seguenti predicati: **`attore(X)`** (X è un attore), **`invitato(X)`** (X è invitato), **`inRitardo(X)`** (X è in ritardo), **`inOrario(X)`** (X è in orario).

Si usi poi il principio di risoluzione per dimostrare che: non c'è alcun attore invitato al party.

Esercizio 2 (5 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 l'algoritmo *min-max* e l'algoritmo *alfa-beta* risolvono il problema e la mossa selezionata dal giocatore.



Esercizio 3 (6 punti)

Dato il seguente programma Prolog:

```
prodotto(_,[],[]).
prodotto(X,[X|T],T1):- !, prodotto(X,T,T1).
prodotto(X,[Y|T],[H|T1]):- H is Y*X, prodotto(X,T,T1).
```

si disegni l'albero SLDNF relativo al goal:

```
?- prodotto(3,[1,3,5], L).
```

Esercizio 4 (4 punti)

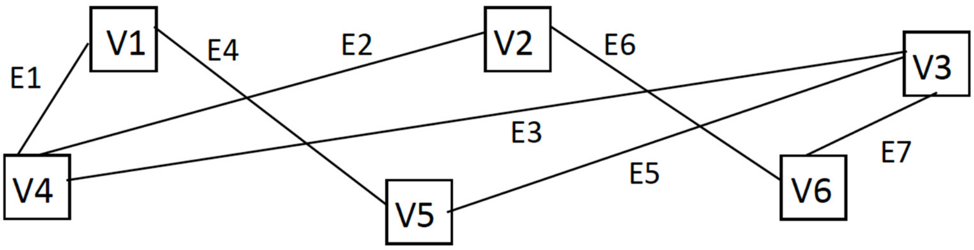
Data una lista L1 di interi si definisca un predicato Prolog **`quadrati`** che riporta la lista L2 dei numeri quadrati presenti in L1. Si supponga di avere a disposizione un predicato **`isSquare/1`** che ha successo se il suo argomento è legato a un valore intero quadrato (ovvero un valore intero che è il quadrato di un altro intero).

Esempio:

```
?-quadrati([1,3,4,25,36],L).
L=[1,4,25,36]
?-quadrati([6,3,5,7],L).
L=[]
```

Esercizio 5 (5 punti)

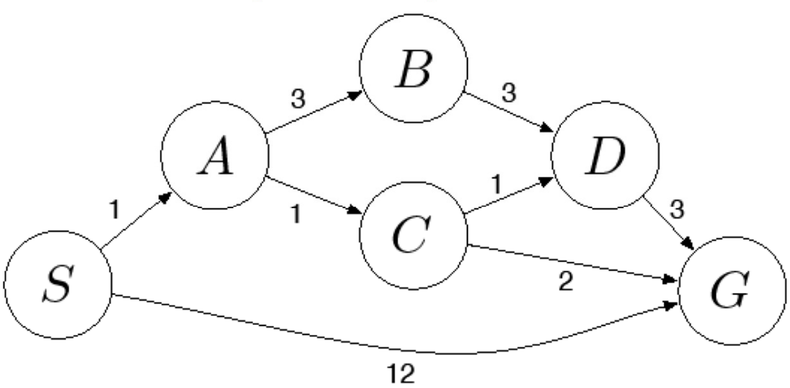
Si consideri il seguente grafo, che rappresenta 6 vertici (*vertexes*: V1-V6), e 7 segmenti (*edges*: E1-E7). In questo problema si può colorare ogni segmento con un colore tra i tre seguenti **{Red, Green, Blue }**.



Si modelli il problema come CSP, le cui variabili sono E1-E7, con il vincolo che due segmenti "adiacenti" non possono essere colorati con lo stesso colore. Ad esempio, E1 e E2 non possono essere colorati con lo stesso colore perché adiacenti in quanto hanno in comune il vertice V4. D'altro canto, E2 e E4 possono avere lo stesso colore in quanto non sono adiacenti (non c'è alcun vertice comune).
Si disegni il constraint graph, e si faccia vedere sui domini l'applicazione del Forward Checking, dopo che la variabile E2 è stata istanziata con il colore **Red**.

Esercizio 6 (6 punti)

Dato il seguente grafo, dove S è lo stato iniziale, G lo stato goal, il costo di ciascun arco è la sua etichetta, e in tabella è riportata la stima euristica della distanza di ciascun nodo dal nodo goal G, si indichi l'ordine con cui la strategia di ricerca A* esplora i nodi alla ricerca di una soluzione. In caso di più nodi figli, a parità di altre condizioni, si espanda per primo quello il cui nome precede alfabeticamente il nome degli altri figli.
Mostrare la soluzione trovata e il suo cammino.
La funzione euristica h(n) introdotta nell'esercizio è ammissibile? E' consistente? Si motivi la risposta data.



State	$h(n)$
S	4
A	2
B	6
C	1
D	3
G	0

FONDAMENTI DI INTELLIGENZA ARTIFICIALE

25 Gennaio 2018 – Soluzioni

Esercizio 1

1. Ogni attore invitato al party arriva in ritardo
2. Chiunque arrivi in orario al party, non arriva in ritardo, e viceversa
3. Tutti gli invitati arrivano in orario al party

Goal: Non c'è alcun attore invitato al party

1. $\forall X$ (attore(X) $\wedge$ invitato(X) $\rightarrow$ inRitardo(X))

2a. $\forall X$ (inOrario(X) $\rightarrow$ $\neg$inRitardo(X))

2b. $\forall X$ (inRitardo(X) $\rightarrow$ $\neg$inOrario(X)) (nota che si poteva anche intendere come:
$\forall X$ ($\neg$ inRitardo(X) $\rightarrow$inOrario(X))

3. $\forall X$ (invitato(X) $\rightarrow$ inOrario(X))

GNeg: $\exists Y$ attore(Y) $\wedge$ invitato(Y)

Clausole:

C1: $\neg$invitato(X) $\vee$ $\neg$ attore(X) $\vee$ inRitardo(X)

C2a: $\neg$inOrario(X) $\vee$ $\neg$inRitardo(X)

C2b: $\neg$inRitardo (X) $\vee$ $\neg$inOrario(X)

C3 $\neg$invitato(X) $\vee$ inOrario(X)

GNeg_a: attore(s) (goal negato, più Skolem)

GNeg_b: invitato(s) (goal negato, più Skolem)

Risoluzione:

C4: GNeg_a+C1: $\neg$invitato(s) $\vee$ inRitardo(s)

C5: C4+GNeg_b: inRitardo(s)

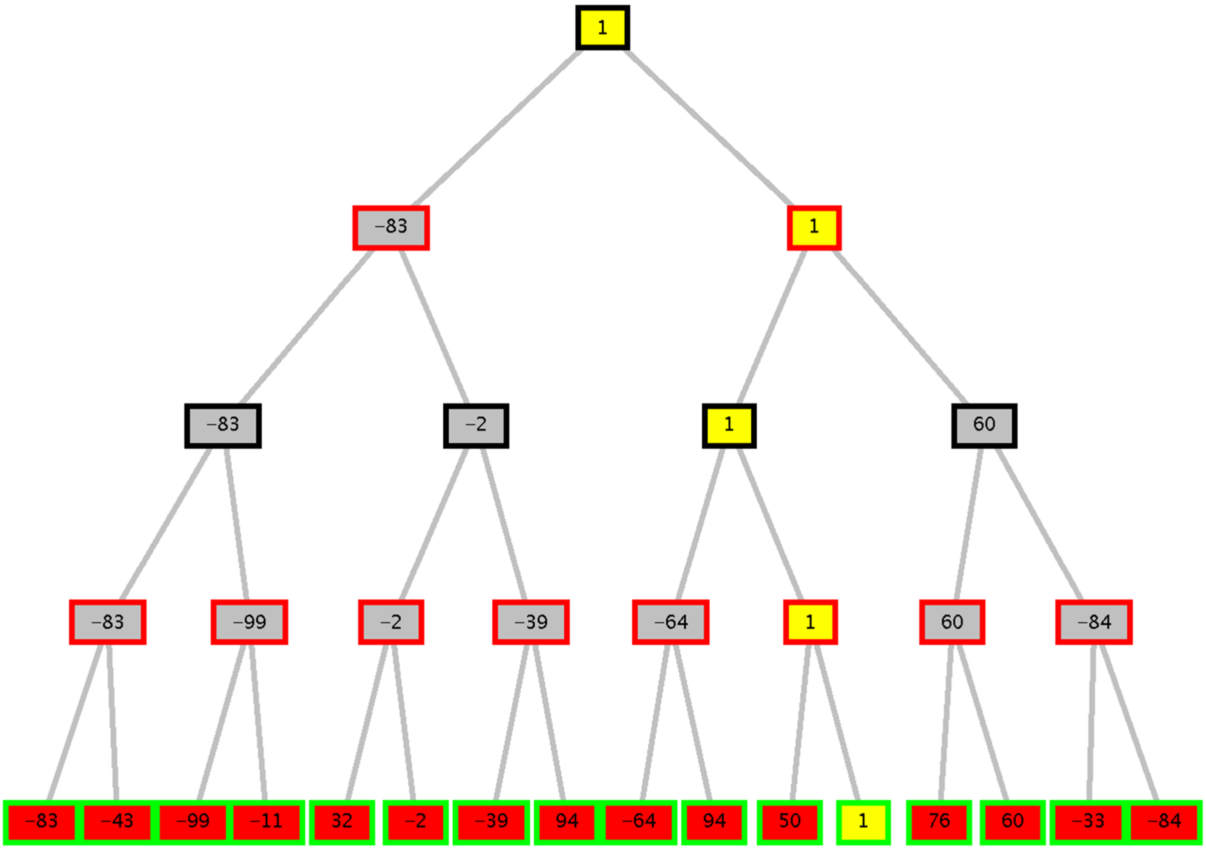
C6: C5+C2a: $\neg$inOrario(s)

C7: C6+C3: $\neg$invitato(s)

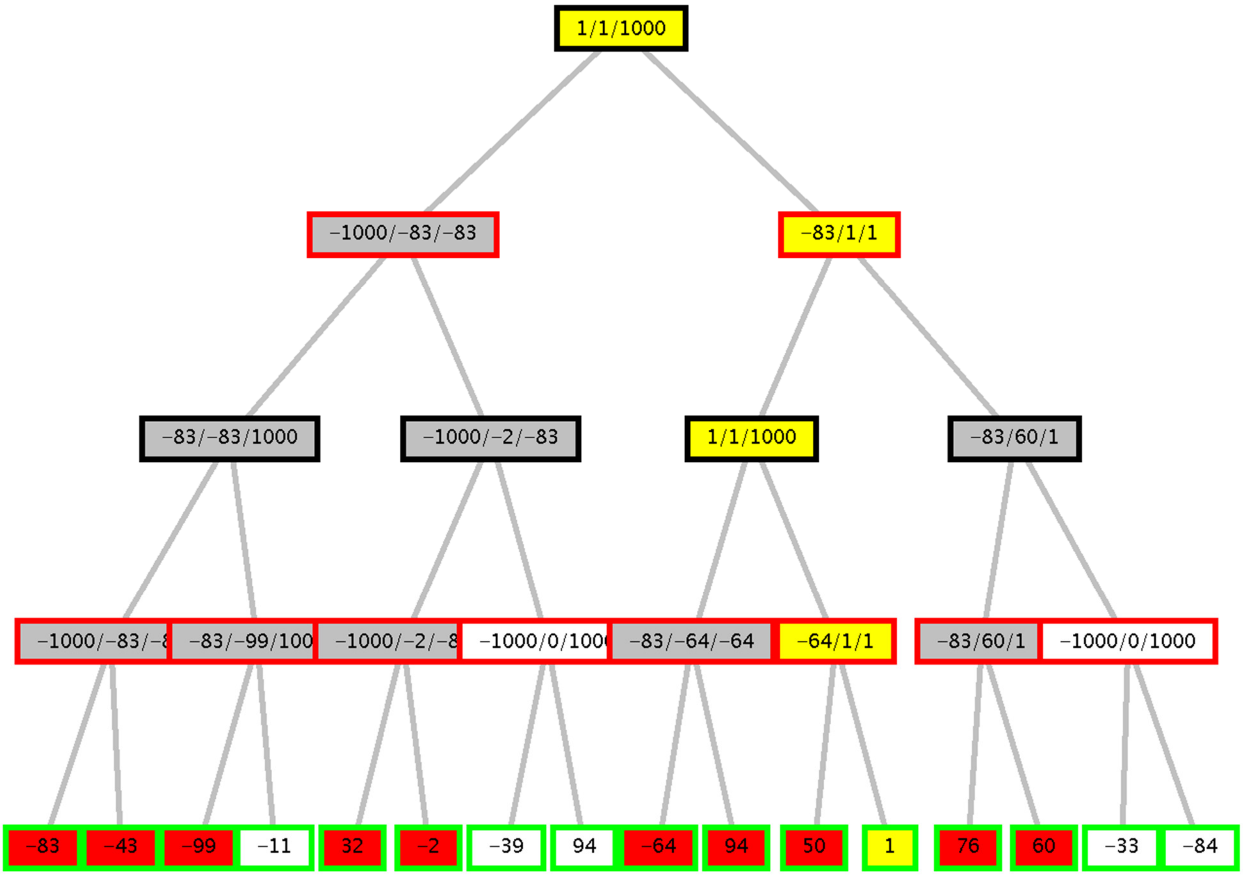
C8: C7+GNeg_b: clausola vuota

Esercizio 2

Min-Max:

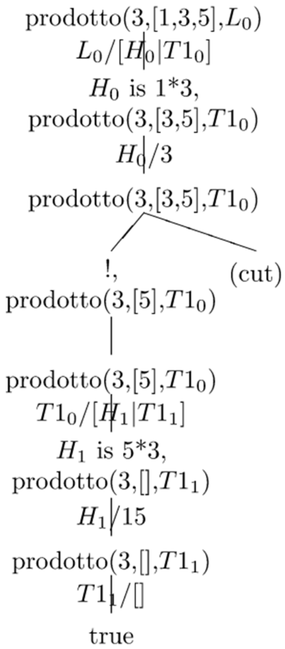


Alfa-beta:



I nodi che portano alla soluzione sono in giallo, quelli tagliati in bianco.

Esercizio 3



Esercizio 4

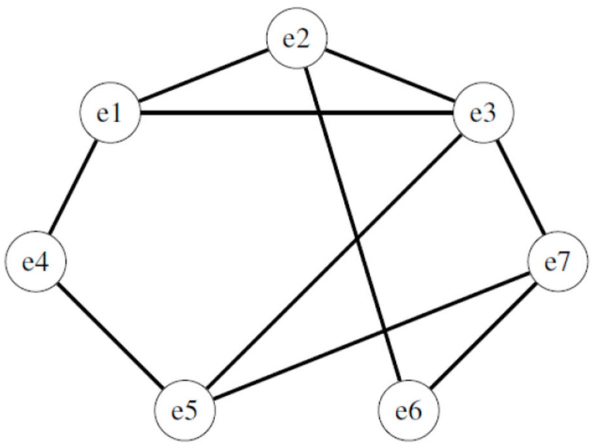
```
quadrati([],[]).
quadrati([X|T],[X|T1]):- isSquare(X),!,quadrati(T,T1).
quadrati([_|T],T1):- quadrati(T,T1).
```

Anche se non richiesto, si riporta di seguito una possibile definizione del predicato isSquare/1 (si noti che float_fractional_part/1 è una funzione definita per SWI-Prolog, SICStus, e in generale per gli interpreti Prolog ISO-compliant).

```
isSquare(X):-  Y is sqrt(X),
               Y1 is float_fractional_part(Y),
               Y1==0.0.
```

Esercizio 5

Vincoli:
E1-E7::[Red,Green,Blu]
E1 $\neq$ E2
E1 $\neq$ E3
E1 $\neq$ E4
E2 $\neq$ E3
E4 $\neq$ E5
E2 $\neq$ E6
E6 $\neq$ E7
E7 $\neq$ E5
E7 $\neq$ E3
E5 $\neq$ E3



Constraint graph (tutti i vincoli/archi sono di diverso $\neq$)

Applicazione del FC:

E1	E2	E3	E4	E5	E6	E7
~~R~~ G B	R	~~R~~ G B	R G B	R G B	~~R~~ G B	R G B

Esercizio 6

A* con $\boldsymbol{h(n)}$.

Ordine di espansione dei nodi : S A C (G)

Soluzione trovata: S A C G

Costo della soluzione trovata: 4

h(n) è ammissibile ma non consistente. Se si considera ad esempio il nodo S h(S) vale 4 che non è <= del costo per andare da S ad A (che vale 1) sommato ad h(A) che vale 2 (risultato 3).