

FONDAMENTI DI INTELLIGENZA ARTIFICIALE

9 Luglio 2021 – Tempo a disposizione: 2 h – Risultato: 32/32 punti

NOTA: Consegnare la soluzione tramite un singolo file denominato con il proprio CognomeNome - Ad esempio: RossiMario

Esercizio 1 (6 punti)

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

1. La professoressa (prof) premia tutti gli studenti appartenenti alle squadre.
2. Ogni studente appartiene a una squadra.
3. Tutti gli studenti premiati dalla professoressa sono bravi.
4. Mario è uno studente.

Si dimostri poi, tramite il principio di risoluzione, che esiste almeno uno studente bravo. Si usino i seguenti predicati con l'ovvio significato:

premia (X, Y) – X premia Y;

studente (X) – X è uno studente;

squadra (X, Y) – X appartiene alla squadra Y;

bravo (X) X è bravo.

I termini "prof", "mario" siano considerati costanti.

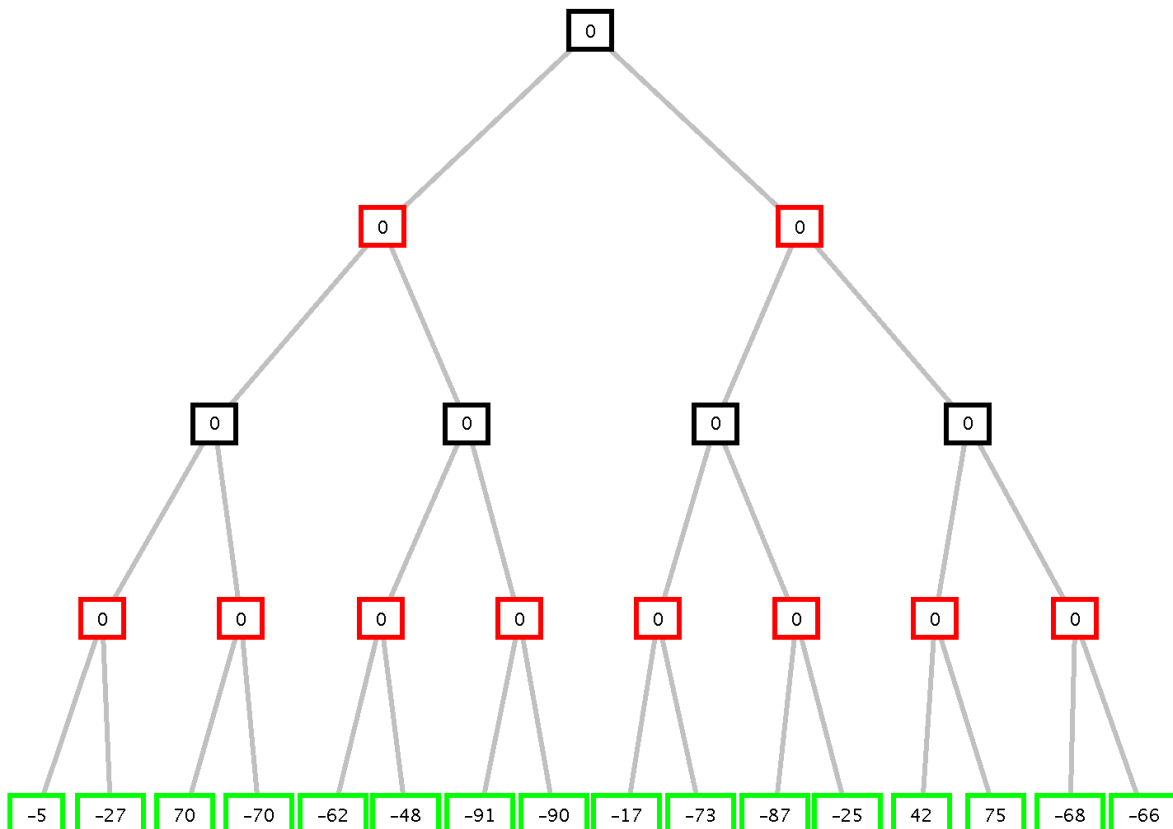
NOTA: Si riportano i simboli degli operatori e quantificatori in logica: $\forall \exists \wedge \vee \neg \rightarrow$

Esercizio 2 (5 punti)

Si consideri il seguente albero di gioco in cui il primo giocatore è MAX.

a) Si indichi come l'algoritmo min-max risolve il problema indicando il valore con cui viene etichettato il nodo iniziale e la mossa selezionata dal primo giocatore (arco a1, o a2).

b) Si mostrino poi i tagli che l'algoritmo alfa-beta consente indicando gli archi che verranno tagliati. Si indichino i nomi degli archi iniziando con la lettera "a" e facendola seguire con un numero crescente da sinistra a destra e dall'alto al basso. Ad esempio, i due archi che si dipartono dalla radice saranno nominati a1 (quello più a sinistra) e a2. L'arco che connette il nodo foglia più a sinistra (con valore -5) sarà denominato a15, mentre l'ultimo arco che connette il nodo foglia più a destra (valore -66) a30.



Esercizio 3 (7 punti)

Si consideri il seguente CSP che lega le variabili A, B, C, D:

A::[1, 2, 3, 4, 5, 6]

B::[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

C::[1, 2, 3, 4, 5, 6]

D::[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

$A \geq B - 8$

$C = B - 5$

$A \leq D - 5$

Si applichi, durante la ricerca fino alla prima soluzione, il Forward Checking dopo ogni passo di labeling, con le due euristiche su scelta della prossima variabile da istanziare:

- Si applichi labeling e FC, considerando le variabili secondo il loro **ordinamento alfabetico**;
- Si applichi labeling e FC, considerando, nella scelta della prossima variabile da istanziare, l'euristica **Minimum Remaining Value** (poi, a parità di cardinalità di dominio, scegliere in base all'ordine alfabetico dei nomi delle variabili). Quanto cambia la ricerca e i passi svolti in questo caso rispetto al precedente a)?

Nel labeling, per il valore da assegnare alla variabile, si considerino in entrambi i casi i valori di dominio in ordine crescente, partendo dal più piccolo.

Esercizio 4 (4 punti)

Si realizzi un predicato Prolog `buildList(X, Y, N)` che dato un elemento X e un numero naturale N, dia in uscita una lista Y in cui l'elemento X è ripetuto N volte. Di seguito si riporta qualche esempio di esecuzione Prolog:

```
?- buildList(a, Y, 0).
```

```
Y = []
```

```
?- buildList(a, Y, 3).
```

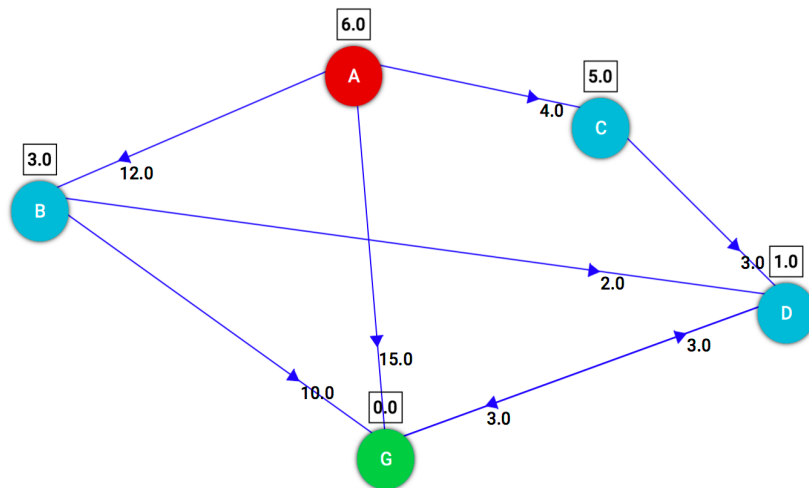
```
Y = [a, a, a]
```

```
?- buildList(a, Y, 1).
```

```
Y = [a]
```

Esercizio 5 (6 punti)

Si consideri il seguente grafo, dove A è il nodo iniziale e G il nodo goal, e il numero associato agli archi è il costo dell'operatore per andare dal nodo di partenza al nodo di arrivo dell'arco. Vicino ad ogni nodo, in un quadrato, è indicata inoltre la stima euristica della sua distanza dal nodo goal G:



- Si applichi la ricerca **A*** su alberi (che non tiene traccia dei nodi già visitati) e si indichino i nodi espansi nell'ordine di espansione. In caso di non-determinismo si scelga il nodo da espandere in base all'ordine alfabetico del nome.
- $h(n)$ è ammissibile? Si motivi la risposta.
- Si indichi il costo della soluzione, motivando se è garantita o meno l'ottimalità.

Esercizio 6 (4 punti)

Si descrivano le tecniche di node-, arc- e n-consistenza su un grafo CSP.

9 luglio 2021 - Soluzioni

Esercizio 1

Trasformazione in clausole

1. $\forall X \forall Y \text{ studente}(X) \wedge \text{squadra}(X,Y) \rightarrow \text{premia}(\text{prof}, X).$
 $\neg \text{studente}(X) \vee \neg \text{squadra}(X,Y) \vee \text{premia}(\text{prof}, X)$
2. $\forall X \text{ studente}(X) \rightarrow \exists S \text{ squadra}(X, S).$
 $\forall X \exists S \neg \text{studente}(X) \vee \text{squadra}(X, f(X))$ Skolem
3. $\forall X \forall Y, \text{ studente}(X) \wedge \text{premia}(\text{prof}, X) \rightarrow \text{bravo}(X).$
 $\neg \text{studente}(X) \vee \neg \text{premia}(\text{prof}, X) \vee \text{bravo}(X)$
4. $\text{studente}(\text{mario})$

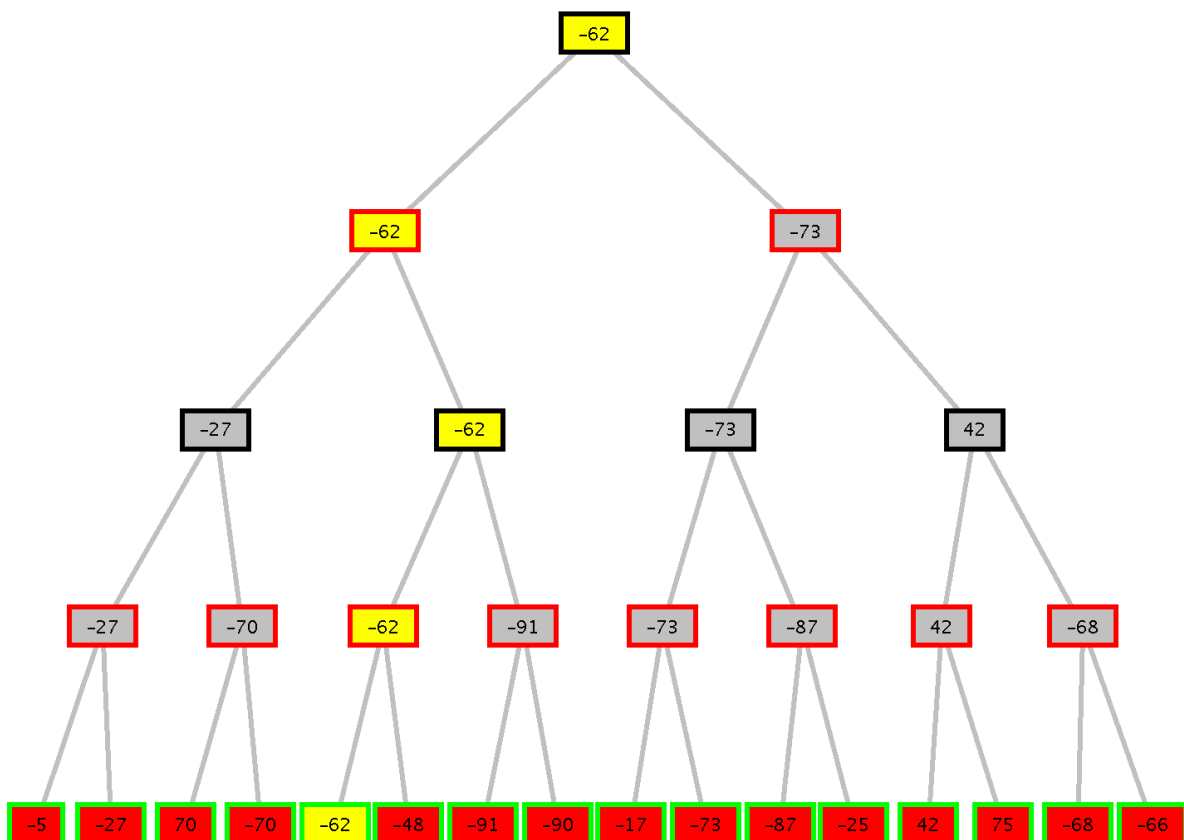
- G. $\exists X \text{ studente}(X) \wedge \text{bravo}(X)$
 Gneg. $\neg \text{studente}(X) \vee \neg \text{bravo}(X)$

Risoluzione

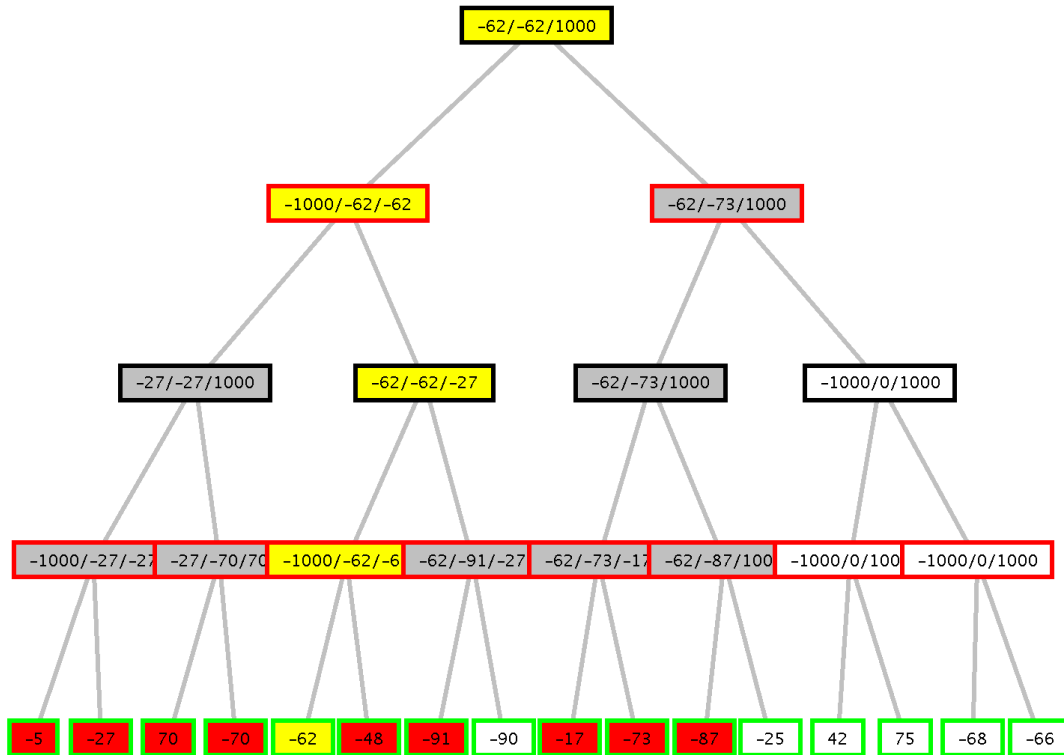
- 5: Gneg + 4: $\neg \text{bravo}(\text{mario})$
- 6: 5 + 3: $\neg \text{studente}(\text{mario}) \vee \neg \text{premia}(\text{prof}, \text{mario})$
- 7: 6 + 4: $\neg \text{premia}(\text{prof}, \text{mario})$
- 8: 7 + 1: $\neg \text{studente}(\text{mario}) \vee \neg \text{squadra}(\text{mario}, X)$
- 9: 8 + 4: $\neg \text{squadra}(\text{mario}, X)$
- 10: 9 + 2: $\neg \text{studente}(\text{mario})$
- 11: 9 + 4: *contraddizione []. Dimostrato*

Esercizio 2

Min-max: strada in giallo – valore nodo radice -62, ramo a1.



In rosso i nodi espansi, in giallo la strada trovata, i nodi in bianco non sono esplorati per effetto dei tagli alfa-beta.



Archi tagliati a6, a22, a26.
 Scelta per il ramo a1, valore propagato -62.

Esercizio 3

Con scelta variabili in base al nome:

	A	B	C	D
Labeling	A=1	[1...10]	[1...6]	[1...10]
FC	A=1	[1...9]	[1...6]	[6...10]
Labeling	A=1	B=1	[1...6]	[6...10]
FC e Backtracking	A=1	B=1	Fail	[6...10]
Labeling	A=1	B=2	[1...6]	[6...10]
FC e Backtracking	A=1	B=2	Fail	[6...10]
Labeling	A=1	B=3	[1...6]	[6...10]
FC e Backtracking	A=1	B=3	Fail	[6...10]
Labeling	A=1	B=4	[1...6]	[6...10]
FC e Backtracking	A=1	B=4	Fail	[6...10]
Labeling	A=1	B=5	[1...6]	[6...10]
FC e Backtracking	A=1	B=5	Fail	[6...10]
Labeling	A=1	B=6	[1...6]	[6...10]
FC	A=1	B=6	[1]	[6...10]
Labeling e FC	A=1	B=6	C=1	[6...10]
Labeling	A=1	B=6	C=1	D=6
Soluzione	A=1	B=6	C=1	D=6

Con euristica MRV:

	A	B	C	D
Labeling	A=1	[1...10]	[1...6]	[1...10]
FC	A=1	[1...9]	[1...6]	[6...10]
Labeling	A=1	[1...9]	[1...6]	D=6
FC	A=1	[1...9]	[1...6]	D=6
Labeling e FC	A=1	[1...9]	C=1	D=6
FC	A=1	[6]	C=1	D=6
Labeling	A=1	B=6	C=1	D=6
Soluzione	A=1	B=6	C=1	D=6

Con euristica MRV si arriva direttamente alla prima soluzione, senza backtracking.

Esercizio 4

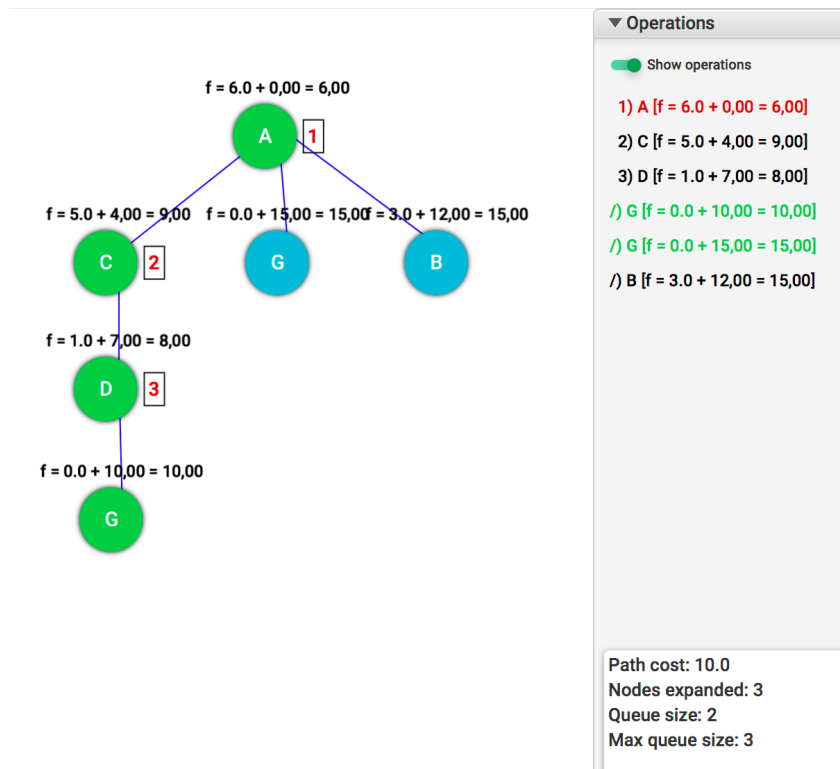
```
buildList(_, [], 0):-!.  
buildList(X,[X|Ys],N) :- N1 is N- 1, buildList(X,Ys,N1).
```

Esercizio 5

Con nodi espansi ACD si trova la soluzione (ottimale):

ACDG con costo 10.

Con A*, la soluzione trovata è ottimale perché l'euristica è ammissibile.



Esercizio 6

Si vedano le slide del corso.