

FONDAMENTI DI INTELLIGENZA ARTIFICIALE

17 Febbraio 2022 – Tempo a disposizione: 2 h – Risultato: 32/32 punti

Esercizio 1 (6 punti)

Si formalizzino le seguenti frasi in logica dei predicati:

1. Tutti i partecipanti alla festa sono vestiti di bianco o di nero (OR non esclusivo).
2. Esiste almeno un partecipante alla festa.
3. Non vi è alcun partecipante alla festa che sia vestito di bianco.

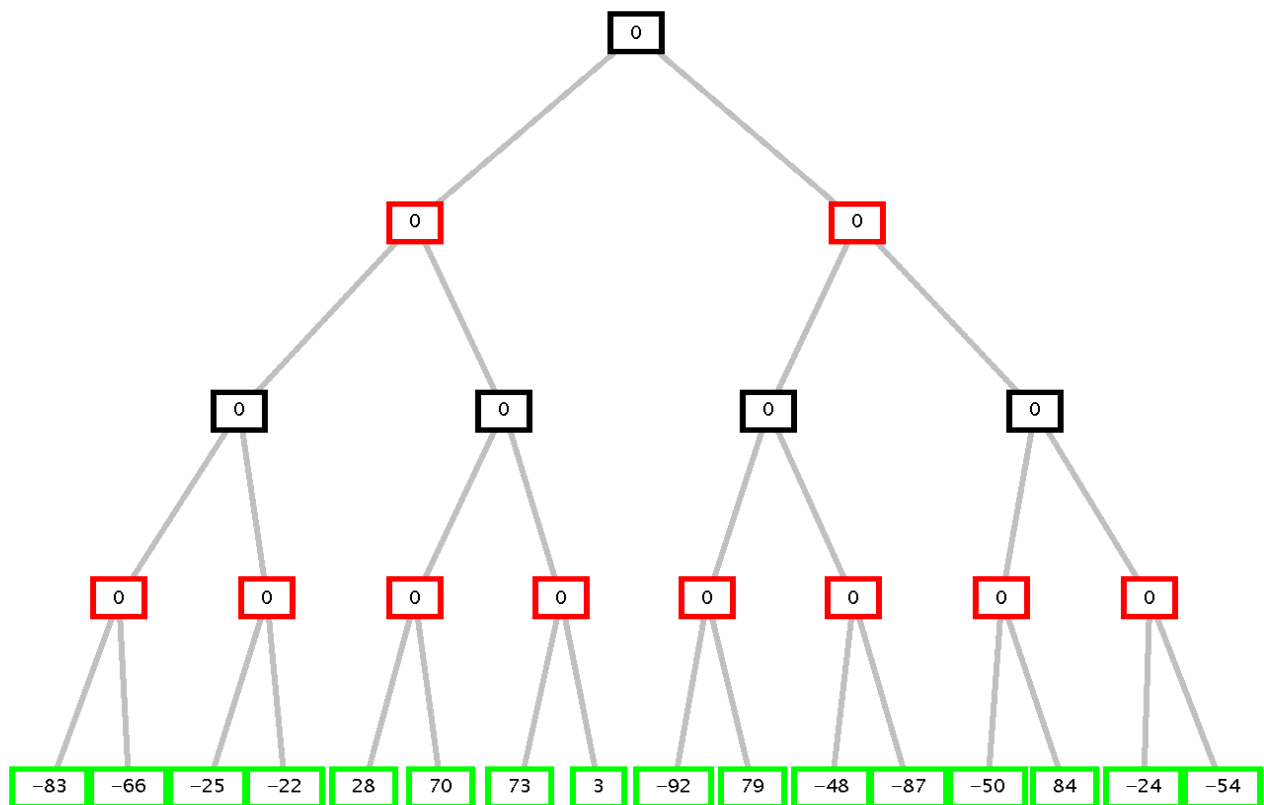
Si utilizzino i predicati **partefesta**(X), **colorevestito**(X,Y) che indicano rispettivamente “X è partecipante alla festa” e “il colore del vestito di X è Y”, e le costanti **bianco** e **nero**. Si trasformino le frasi in clausole e poi si utilizzi il principio di risoluzione per derivare il teorema “vi è un partecipante alla festa che è vestito di nero”.

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 -83) sarà denominato a15, mentre l'ultimo arco che connette il nodo foglia più a destra (valore -54) a30.



Esercizio 3 (5 punti)

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

A:[4, 5, 6]	A<=B+3
B:[2, 3, 4, 5]	C<B-2
C:[2, 3, 4, 5]	D<A+4
D:[2, 3, 4, 5]	

Si applichi, durante la ricerca fino alla prima soluzione, il Forward Checking dopo ogni passo di labeling, 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). Nel labeling, per il valore da assegnare alla variabile, si considerino i valori di dominio in ordine crescente, partendo dal più piccolo.

Esercizio 4 (5 punti)

Si scriva un programma Prolog **listint(List1,List2,Lista3)** che date due liste, **List1**, composta a sua volta da liste di interi, e **List2** composta da numeri interi, restituisca in uscita una lista **List3** che contiene tutti gli elementi di **List2** che non appartengono a nessuna delle liste in **List1**. Se **List1** è vuota restituirà la lista **List2**. Se **List2** è vuota restituirà la lista vuota. Ad esempio:

```
?- listint([[4,5],[3], [1,9], []], [10, 4, 1, 7], List3)
```

```
yes List3=[10,7].
```

```
?- listint([], [10, 4, 1, 7], List3)
```

```
yes List3=[10,4,1,7].
```

```
?- listint([[4,5],[3], [1,9], []], [], List3)
```

```
yes List3=[].
```

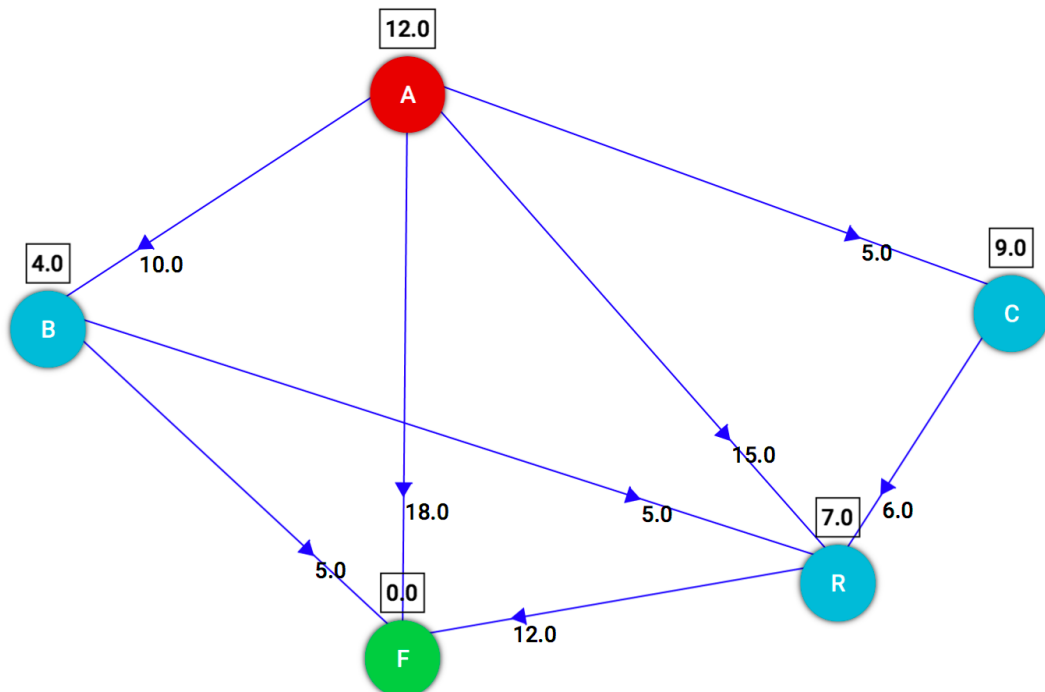
Per la soluzione si può anche utilizzare il predicato **member/2** così definito:

```
member(X,[X|_]).
```

```
member(X,[_|R]) :- member(X,R).
```

Esercizio 5 (7 punti)

Si consideri il seguente grafo, dove A è il nodo iniziale e F 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 F:



Si applichi la ricerca **A*** su alberi (non tenendo quindi traccia dei nodi già visitati) indicando:

- i nodi espansi nell'ordine di espansione. In caso di non-determinismo, si scelgano i nodi da espandere in base all'ordine alfabetico del loro nome;
- i nodi sulla strada della soluzione e il costo della soluzione;
- se è garantita o meno l'ottimalità e perchè.

Esercizio 6 (4 punti)

Si enuncino a parole e in modo formale le proprietà di correttezza e completezza per un sistema logico.

Esercizio 1

Traduzione in logica

1. $\forall X (\text{partfesta}(X) \Rightarrow \text{colorevestito}(X, \text{bianco}) \vee \text{colorevestito}(X, \text{nero}))$
2. $\exists X \text{partfesta}(X)$
3. $\neg (\exists X (\text{partfesta}(X) \wedge \text{colorevestito}(X, \text{bianco})))$

Goal: $\exists X (\text{partfesta}(X) \wedge \text{colorevestito}(X, \text{nero}))$

Goal negato: $\neg (\exists X (\text{partfesta}(X) \wedge \text{colorevestito}(X, \text{nero})))$

Trasformazione in clausole

1. $\neg \text{partfesta}(X) \vee \text{colorevestito}(X, \text{bianco}) \vee \text{colorevestito}(X, \text{nero})$
2. $\text{partfesta}(c1)$ costante di Skolem
3. $\neg \text{partfesta}(X) \vee \neg \text{colorevestito}(X, \text{bianco})$
4. Gneg: $\neg \text{partfesta}(X) \vee \neg \text{colorevestito}(X, \text{nero})$

Risoluzione

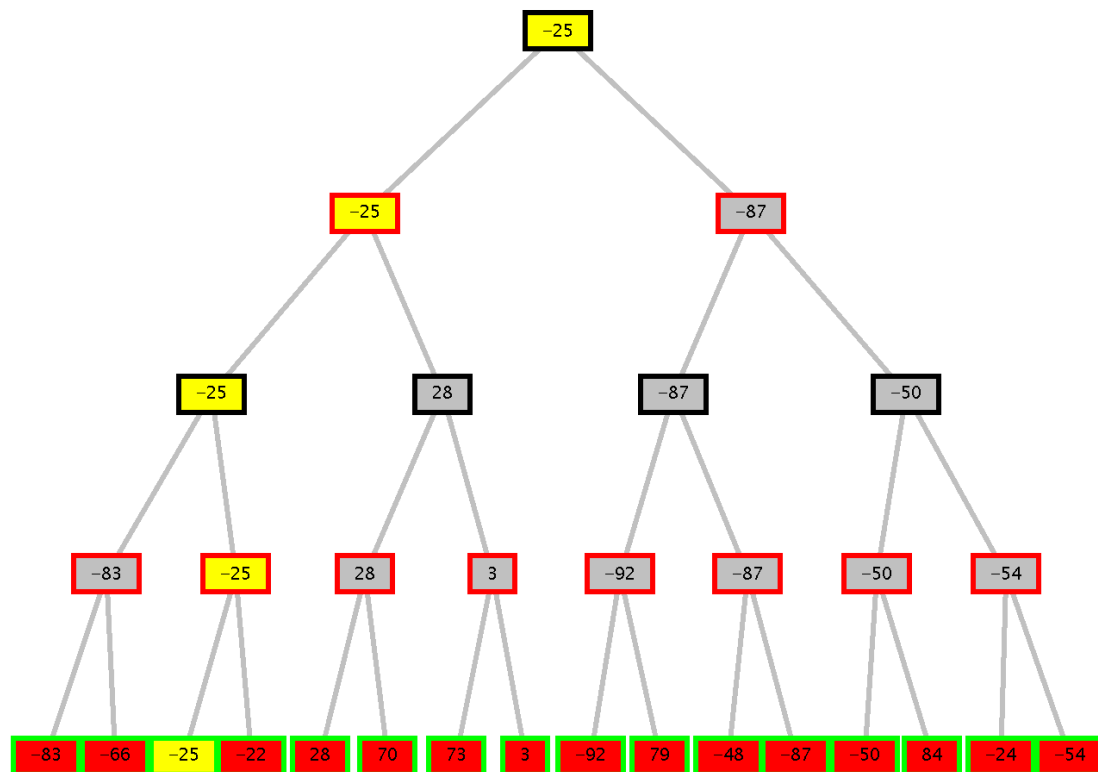
5. (da 2, 4) $\neg \text{colorevestito}(c1, \text{nero})$
6. (da 1, 5) $\neg \text{partfesta}(c1) \vee \text{colorevestito}(c1, \text{bianco})$
7. (da 6, 3) $\neg \text{partfesta}(c1)$
8. (da 7, 2) Contraddizione, Clausola vuota.

Risoluzione alternativa

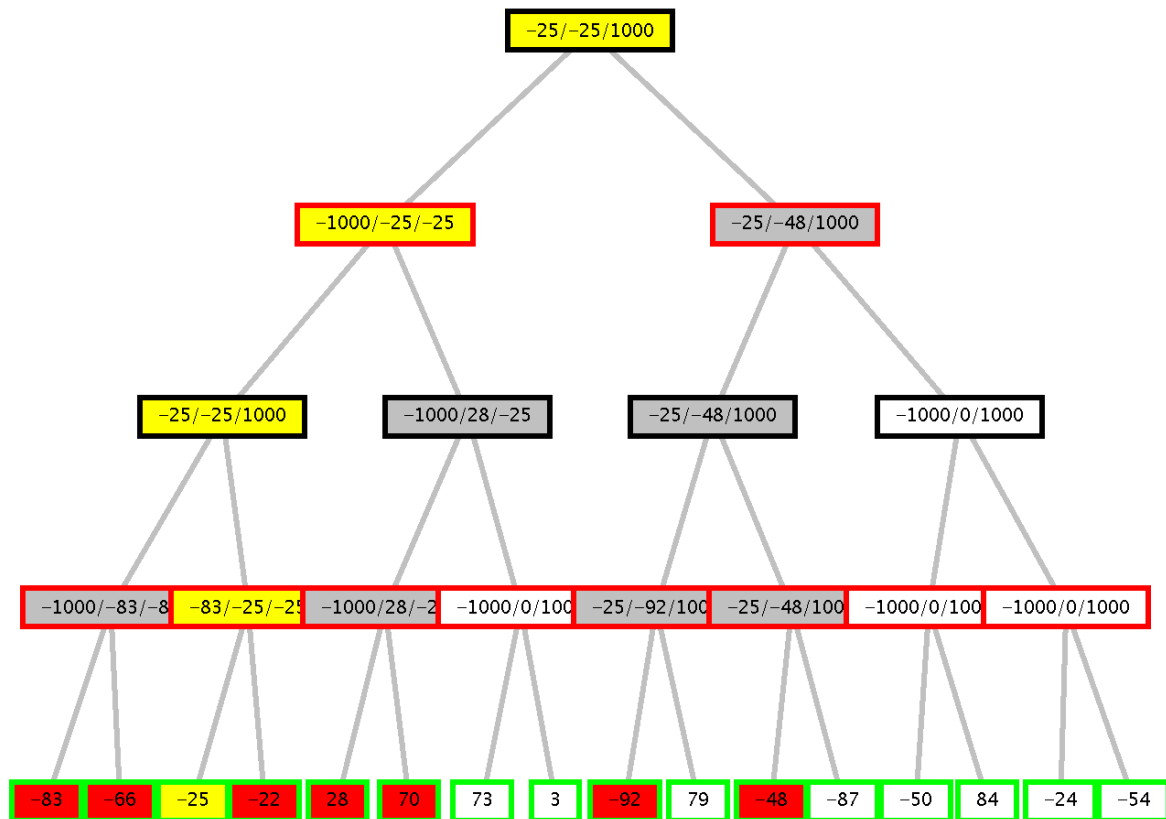
5. (da 1, 4) $\neg \text{partfesta}(X) \vee \text{colorevestito}(X, \text{bianco})$
6. (da 3, 5) $\neg \text{partfesta}(X)$
7. (da 6, 2) Clausola vuota, contraddizione.

Esercizio 2

Min-max: strada in giallo, valore nodo radice -25, ramo a1.



Alfa-Beta: 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 a10, a24, a26, a6 (4 tagli).

Esercizio 3

Con euristica MRV:

	A	B	C	D
Labeling	A=4	[2...5]	[2...5]	[2...5]
FC		[2...5]	[2...5]	[2...5]
Labeling	A=4	B=2	[2...5]	[2...5]
FC	A=4	B=2	[]	[2...5]
Backtracking			Fail	
Labeling	A=4	B=3	[2...5]	[2...5]
FC	A=4	B=3	[]	[2...5]
Backtracking			Fail	
Labeling	A=4	B=4	[2...5]	[2...5]
FC	A=4	B=4	[]	[2...5]
Backtracking			Fail	
Labeling	A=4	B=5	[2...5]	[2...5]
FC	A=4	B=5	[2]	[2...5]
Labeling	A=4	B=5	C=2	[2...5]
FC	A=4	B=5	C=2	[2...5]
Labeling	A=4	B=5	C=2	D=2
Soluzione	A=4	B=5	C=2	D=2

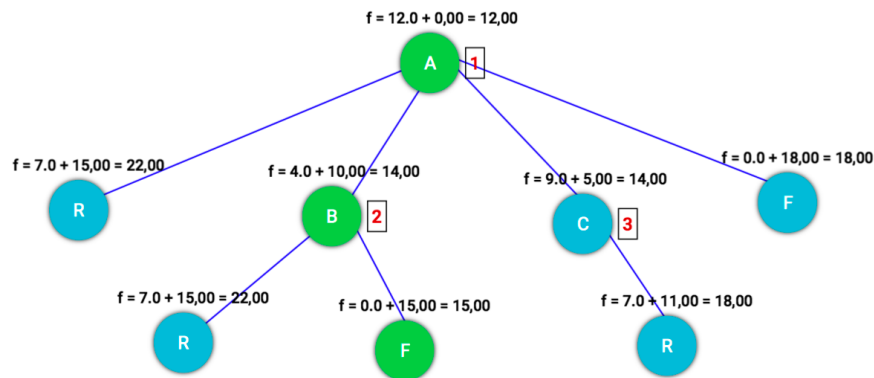
Esercizio 4

```
listint(_, [], []).
listint(L, [A|B], [A|C]) :- not(memberL(A, L)), !, listint(L, B, C).
listint(L, [_|B], L2) :- listint(L, B, L2).
```

```
memberL(X, [L|_]) :- member(X, L), !.
memberL(X, [_|R]) :- memberL(X, R).
```

Esercizio 5

Con A*, i nodi espansi sono ABCF, la soluzione trovata ABF è ottimale perché l'euristica è ammissibile e il costo della soluzione trovata è 15:



Operations

Show operations

- 1) A [$f = 12.0 + 0.00 = 12.00$]
 - 2) B [$f = 4.0 + 10.00 = 14.00$]
 - 3) C [$f = 9.0 + 5.00 = 14.00$]
- /) R [$f = 7.0 + 15.00 = 22.00$]
- /) R [$f = 7.0 + 15.00 = 22.00$]
- /) F [$f = 0.0 + 15.00 = 15.00$]
- /) R [$f = 7.0 + 11.00 = 18.00$]
- /) F [$f = 0.0 + 18.00 = 18.00$]

Path cost: 15.0
Nodes expanded: 3
Queue size: 4
Max queue size: 5

Esercizio 6

Si consulti il materiale del Corso