

Esercizio 1 (6 punti)

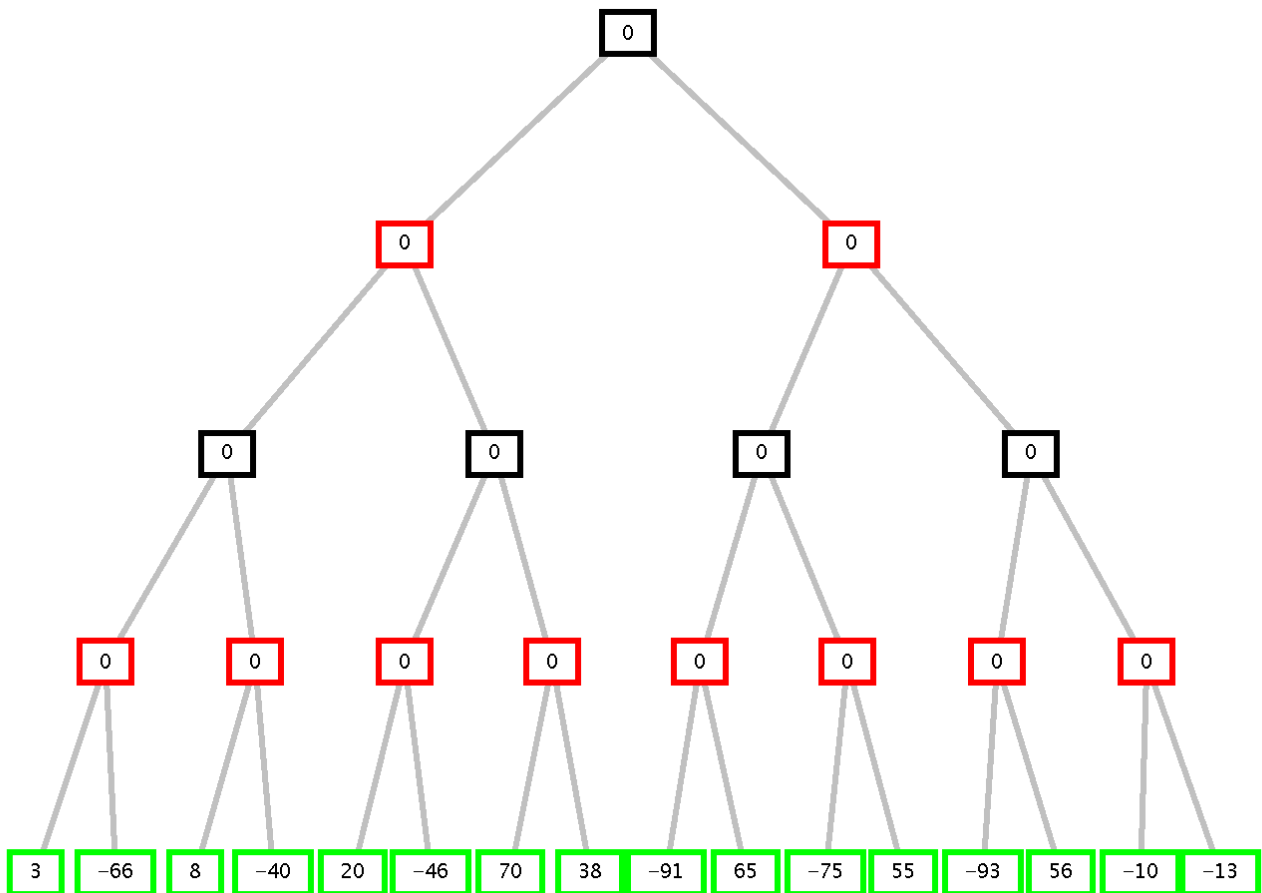
Si formalizzino le seguenti frasi in logica dei predicati:

- Esiste almeno un'aula nel Dipartimento di Ingegneria accessibile
- Ogni aula accessibile ha una postazione per disabili
- Ogni aula che ha una postazione disabili ospita studenti disabili
- Ogni aula che ospita studenti disabili ha una postazione disabili
- L'aula A3 del Dipartimento di Ingegneria è accessibile

Le si trasformi in clausole e si usi poi il principio di risoluzione per dimostrare che c'è un'aula del Dipartimento Ingegneria che ospita disabili.

Esercizio 2 (4 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 (7 punti)

Dato il seguente programma Prolog:

```

sublist([X], [X|_T]).
sublist([X|T], [X|T1]):-!,
    sublist(T, T1).
sublist([X|T], [_X|T1]):-
    sublist([X|T], T1).
    
```

si disegni l'albero SLD relativo al goal:

```
?- sublist([1,3],[2,1,3,5]).
```

Si sviluppi l'albero completo, non fermandosi alla prima soluzione.

Esercizio 4 (5 punti)

Data una lista L di liste di coppie di numeri si definisca un predicato Prolog `coppie(L, L1)` in grado di restituire una nuova lista semplice L1 contenente come elementi i numeri risultanti dalla somma delle coppie di L che hanno il primo elemento strettamente maggiore del secondo, ignorando le altre.

Esempio:

```
?- coppie([[1,2],[7,2],[2,2],[9,5]], L1).
```

```
Yes L1 = [9, 14].
```

Esercizio 5 (6 punti)

Formulare il seguente problema come CSP, individuando variabili, domini e vincoli unari e binari:

“Alice, Barbara, Carla e Daria sono 4 bambine di età (espressa in anni) compresa fra 1 e 10 anni. Esse possiedono 4 cagnolini, rispettivamente Rococò, Sibilo, Tocco e Uguccione, anch'essi di età (espressa in anni) compresa fra 1 e 10 anni. L'età di ciascuna padroncina è il doppio di quella del proprio cane. Alice ha l'età del cane di Barbara.”

Si applichi poi l'algoritmo AC3, mostrando se e come sono ridotti i domini delle variabili.

E' possibile identificare già una soluzione dopo l'applicazione di tale algoritmo?

Esercizio 6 (4 punti)

Si descriva cos'è l'unificazione e il problema relativo all'assenza di occur check nel linguaggio Prolog.

Esercizio 1

- Esiste almeno un'aula nel Dipartimento di Ingegneria accessibile
 $\exists Y(\text{aulaIng}(Y) \wedge \text{accessibile}(Y))$
- Ogni aula accessibile ha una postazione per disabili
 $\forall X(\text{accessibile}(X) \rightarrow \text{haPostazioneDis}(X))$
- Ogni aula che ha una postazione disabili ospita studenti disabili
 $\forall X(\text{haPostazioneDis}(X) \rightarrow \text{ospitaDis}(X))$
- Ogni aula che ospita studenti disabili ha una postazione disabili
 $\forall X(\text{ospitaDis}(X) \rightarrow \text{haPostazioneDis}(X))$
- L'aula A3 del Dipartimento di Ingegneria è accessibile
 $\text{aulaIng}(a3) \wedge \text{accessibile}(a3)$

Goal: $\exists Y \text{aulaIng}(Y) \wedge \text{ospitaDis}(Y)$

Clausole:

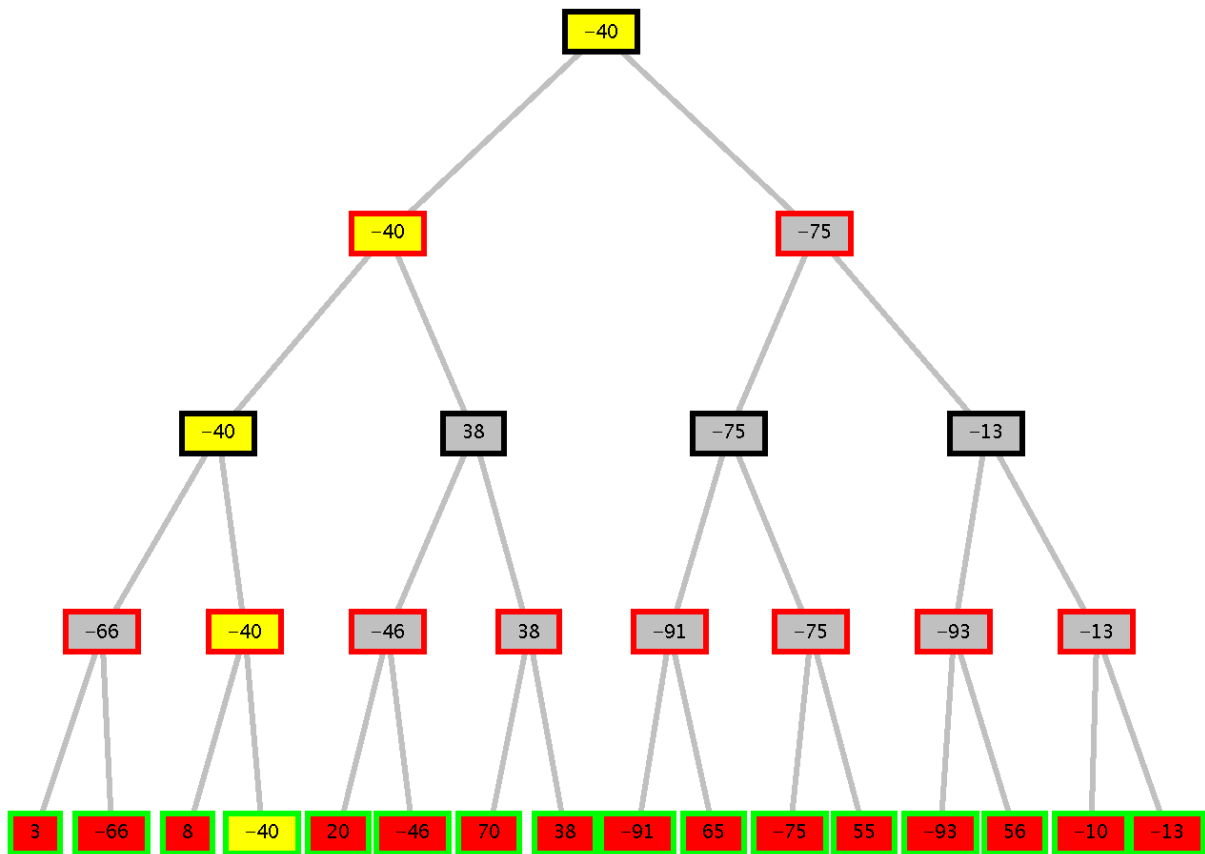
- C1 $\text{aulaIng}(c)$
- C2 $\text{accessibile}(c)$
- C3 $\neg \text{accessibile}(X) \vee \text{haPostazioneDis}(X)$
- C4 $\neg \text{haPostazioneDis}(X) \vee \text{ospitaDis}(X)$
- C5 $\neg \text{ospitaDis}(X) \vee \text{haPostazioneDis}(X)$
- C6 $\text{aulaIng}(a3)$
- C7 $\text{accessibile}(a3)$
- C8 $\neg \text{aulaIng}(Y) \vee \neg \text{ospitaDis}(Y)$ (goal negato)

Risoluzione:

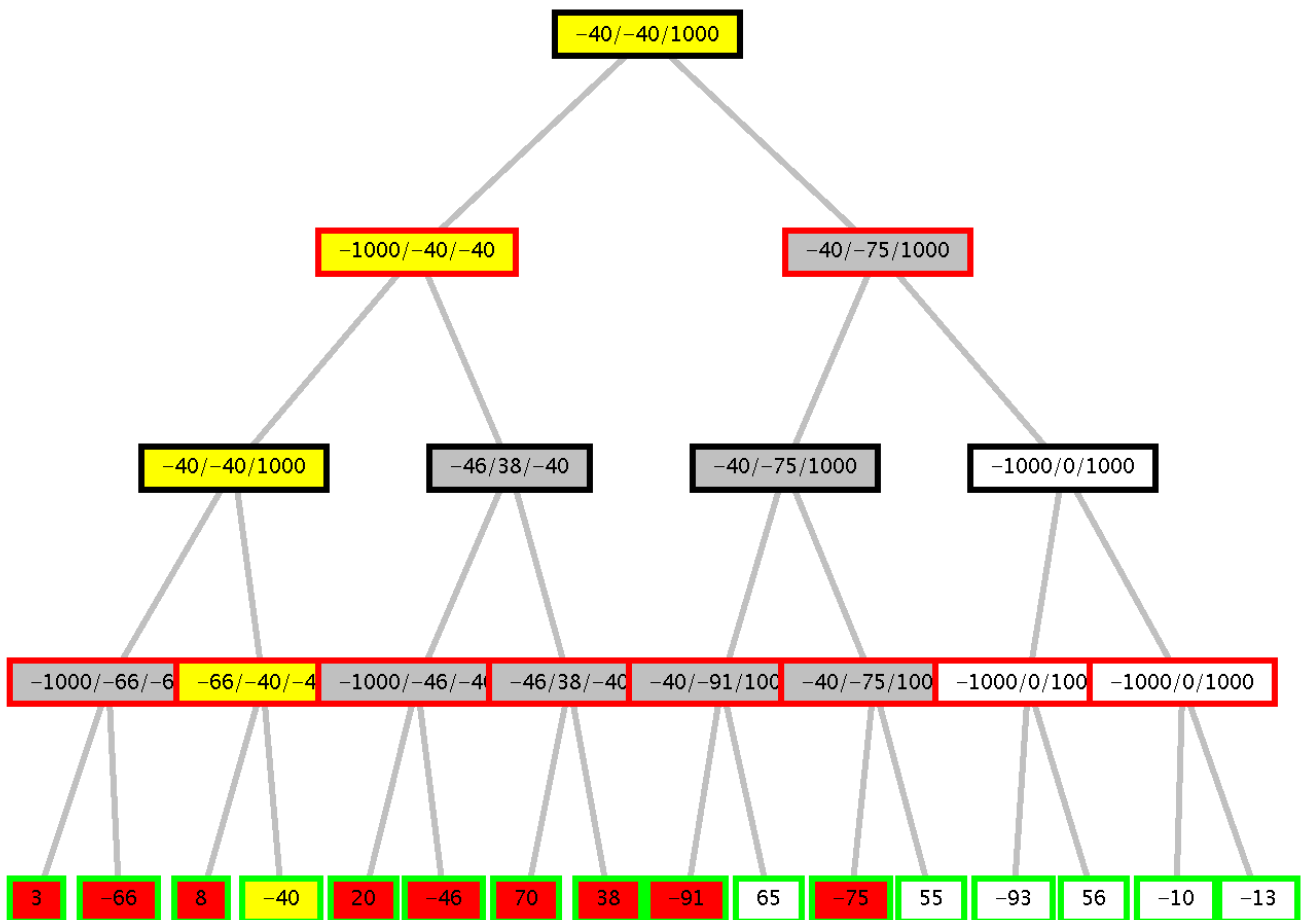
- da C8 e C4: C9 $\neg \text{haPostazioneDis}(Y) \vee \neg \text{aulaIng}(Y)$
- da C9 e C3: C10 $\neg \text{accessibile}(Y) \vee \neg \text{aulaIng}(Y)$
- da C10 e C7: C11 $\neg \text{aulaIng}(a3)$
 (oppure
 da C10 e C6: C11 $\neg \text{accessibile}(a3)$)
- da C11 e C6: C12 contraddizione
 (oppure da C11 e C7: C12 contraddizione)

Si noti che erano corrette anche altre formulazioni che, ad esempio, rendevano esplicito il predicato aula)

Esercizio Min-Max:

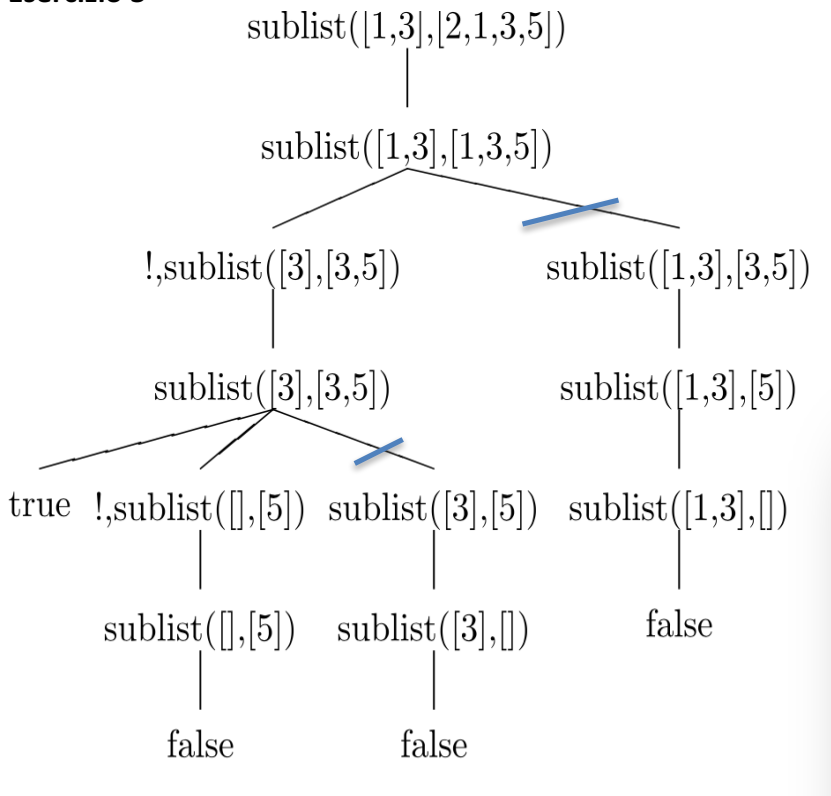


Alfa-beta:



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

Esercizio 3



I rami con il segno  sono tagliati dal cut (!).

Esercizio 4

```
coppie([], []).
coppie([X,Y|RS], [X1|R1S]) :- X > Y,!, X1 is X + Y, coppie(RS, R1S).
coppie([_|RS], R1S) :- coppie(RS, R1S).
```

Esercizio 5

Variabili e Domini:

Variabili A, B, C, D, per le bambine

R, S, T, U per i cani.

DA = DB = DC = DD = [1..10]

DR = DS = DT = DU = [1..10]

Vincoli binari:

L'età di ciascuna padroncina è il doppio di quella del proprio cane.

Alice ha l'età del cane di Barbara.

$A = 2 * R$,

$B = 2 * S$,

$C = 2 * T$,

$D = 2 * U$

$A = S$.

AC3

	A	B	C	D	R	S	T	U
	1..10	1..10	1..10	1..10	1..10	1..10	1..10	1..10
Var1=2*Var2	2,4,6,8,10	2,4,6,8,10	2,4,6,8,10	2,4,6,8,10	1,2,3,4,5	1,2,3,4,5	1,2,3,4,5	1,2,3,4,5
Vincolo A=S	2,4					2,4		
Var1=2*Var2		4,8			1,2			

Non è possibile determinare una soluzione direttamente (i domini delle variabili contengono più valori). Occorrerebbe quindi applicare labeling e poi un algoritmo di propagazione.

Esercizio 6

Vedi slide