

**Esercizio 1 (punti 6)**

Modellare in logica del I ordine le seguenti frasi:

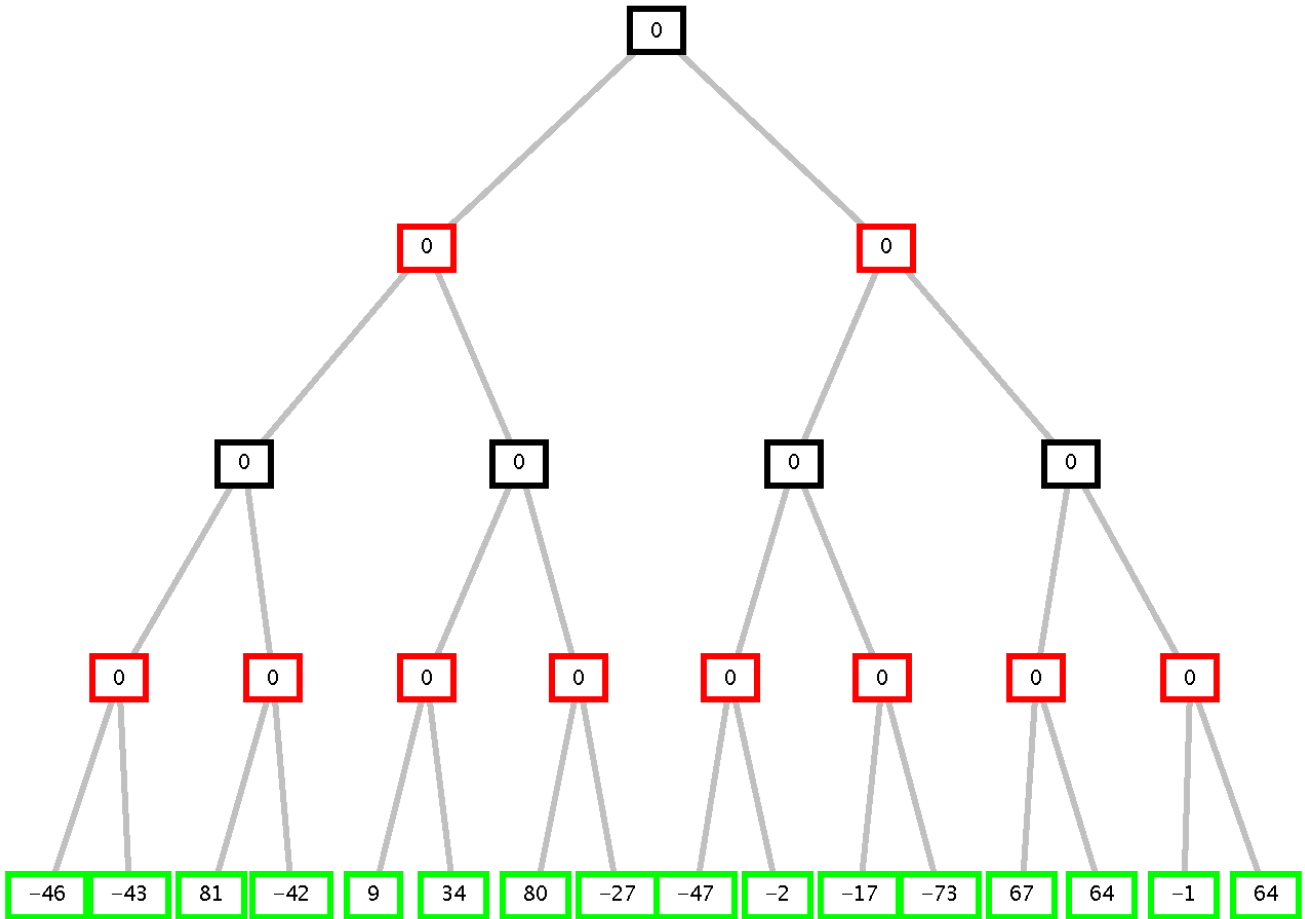
1. Chi ha capacità razionali capisce la logica
2. Chi non ha capacità razionali non capisce la logica
3. Alcuni non capiscono la logica
4. Chi non capisce la logica sbaglia l'esercizio.
5. Cartesio ha capacità razionali
6. Mirco non ha capacità razionali

Si mettano tutte le formule in forma a clausole e si dimostri poi, mediante il principio di risoluzione, che Mirco sbaglia l'esercizio.

Si usino i predicati **logico(X)**, **razionale(X)**, **sbaglia(X)** (con significato: X capisce la logica, X ha capacità razionali, X sbaglia l'esercizio).

**Esercizio 2 (punti 4)**

Si consideri il seguente albero di gioco in cui il primo giocatore è *MAX*. Si mostri come l'algoritmo *min-max* e l'algoritmo *alfa-beta* risolvono il problema e la mossa selezionata dal primo giocatore.



**Esercizio 3 (punti 6)**

Dato il seguente programma Prolog:

```

connesso(X,Y):-arco(X,Y),!.
connesso(X,Y):-arco(X,Z),connesso(Z,Y),!.
connesso(X,Y):-connesso(Y,X).
arco(a,b).
arco(b,c).
arco(b,d).
arco(g,f).
    
```

disegnare l'albero SLD per il goal seguente (si indichino i tagli effettuati dal *cut* e non si espandano i rami tagliati):  
 ?- connesso(c,a).

#### Esercizio 4 (punti 5)

Si definisca un predicato `stessaSomma(L,V)` che data una lista di liste non vuota  $L$ , controlla che le somme di tutti gli elementi per ciascuna delle (sotto-)liste contenute in  $L$  siano uguali a  $V$ .

Ad esempio per il goal:

```
?-stessaSomma([[6,7,2],[1,5,9],[8,3,4]],V).
```

Yes  $V=15$

perché la somma di tutti gli elementi di ciascuna sotto-lista è pari a 15. Altri esempi:

```
?-stessaSomma([[6,4,1,2],[1,12],[8,5]],V).
```

Yes  $V=13$

```
?-stessaSomma([[6,7],[1,12],[8,5]],13).
```

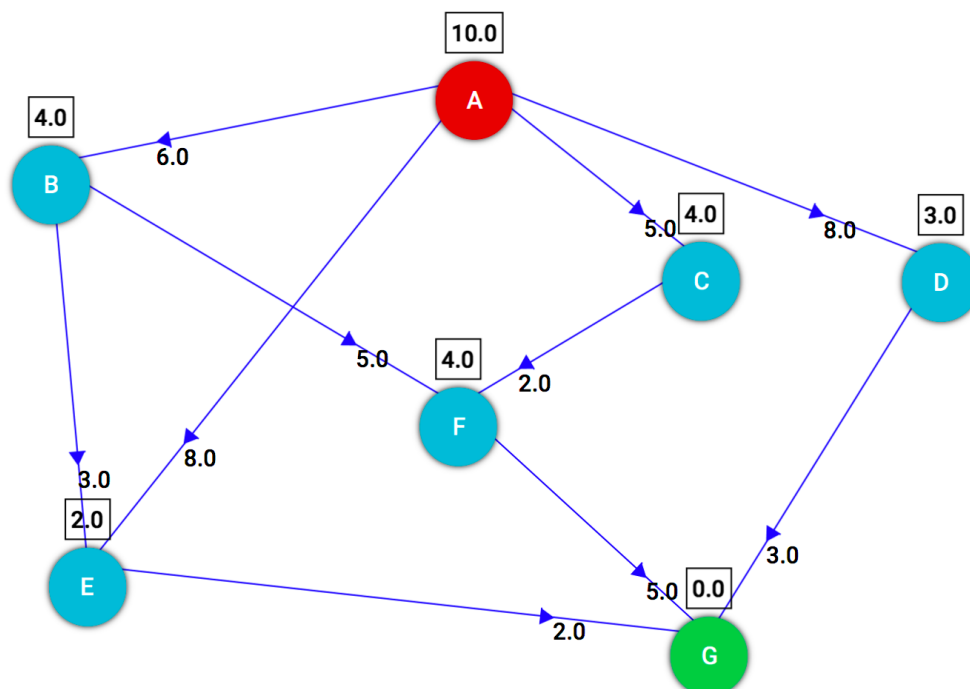
Yes

```
?-stessaSomma([],V).
```

No

#### Esercizio 5 (punti 7)

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. A fianco di ogni nodo, in un quadrato, è indicata inoltre la stima euristica della sua distanza dal nodo goal.



- Si applichi la ricerca depth-first (quindi non si considerino i costi degli archi), e si disegni l'albero di ricerca sviluppato indicando per ogni nodo  $n$  l'ordine di espansione; in caso di non-determinismo, si scelgano i nodi da espandere in base all'ordine alfabetico.
- Si applichi poi la ricerca A\*, e si disegni l'albero di ricerca sviluppato indicando per ogni nodo  $n$  l'ordine di espansione. In caso di non-determinismo, si scelgano i nodi da espandere in base all'ordine alfabetico. Si consideri come euristica  $h(n)$  quella indicata nel quadrato a fianco di ogni nodo in figura (l'euristica è ammissibile). Confrontare le due soluzioni trovate, in termini di costo di cammino. Quale dà la soluzione migliore?

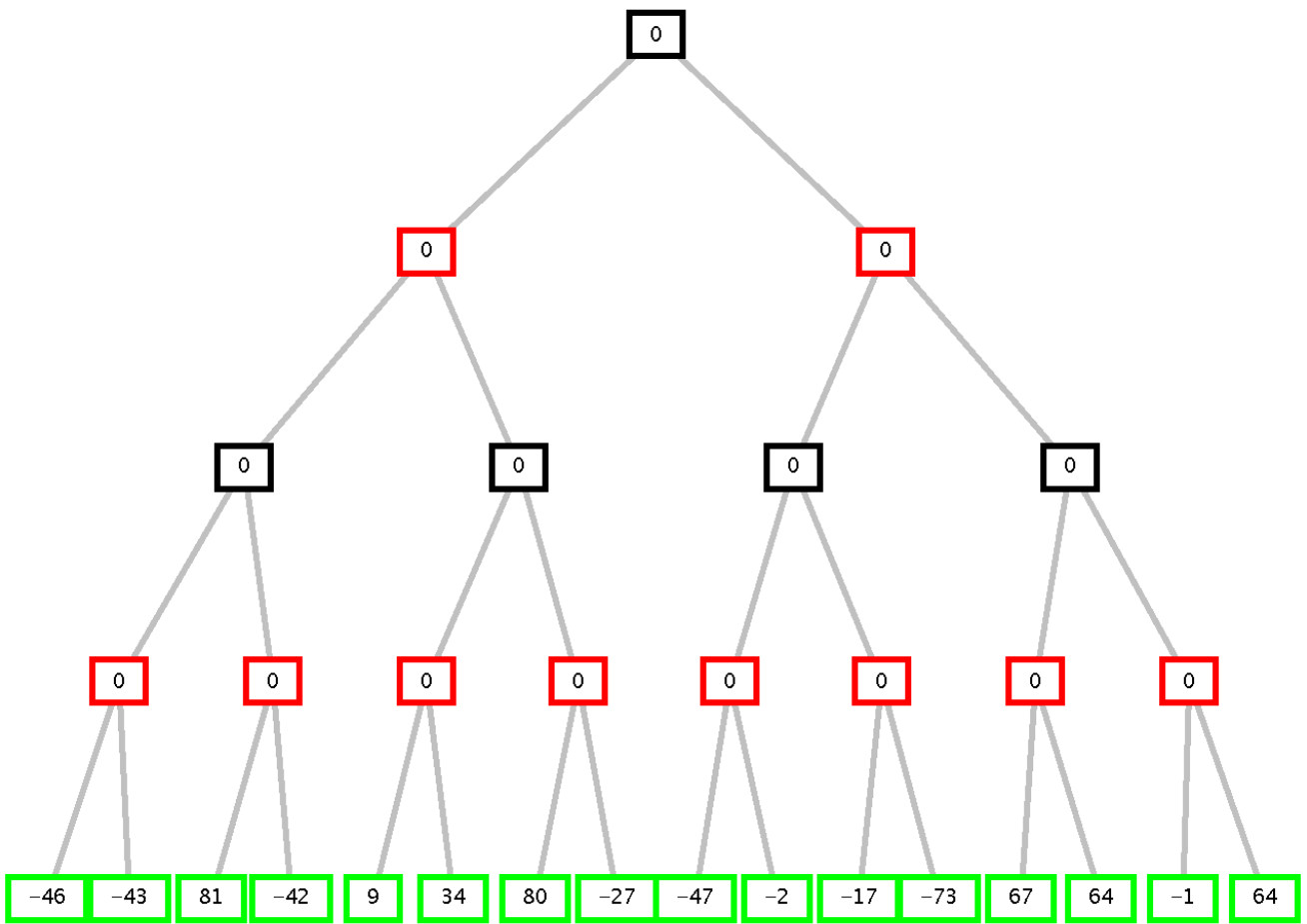
#### Esercizio 6 (punti 4)

Si consideri il seguente CSP:

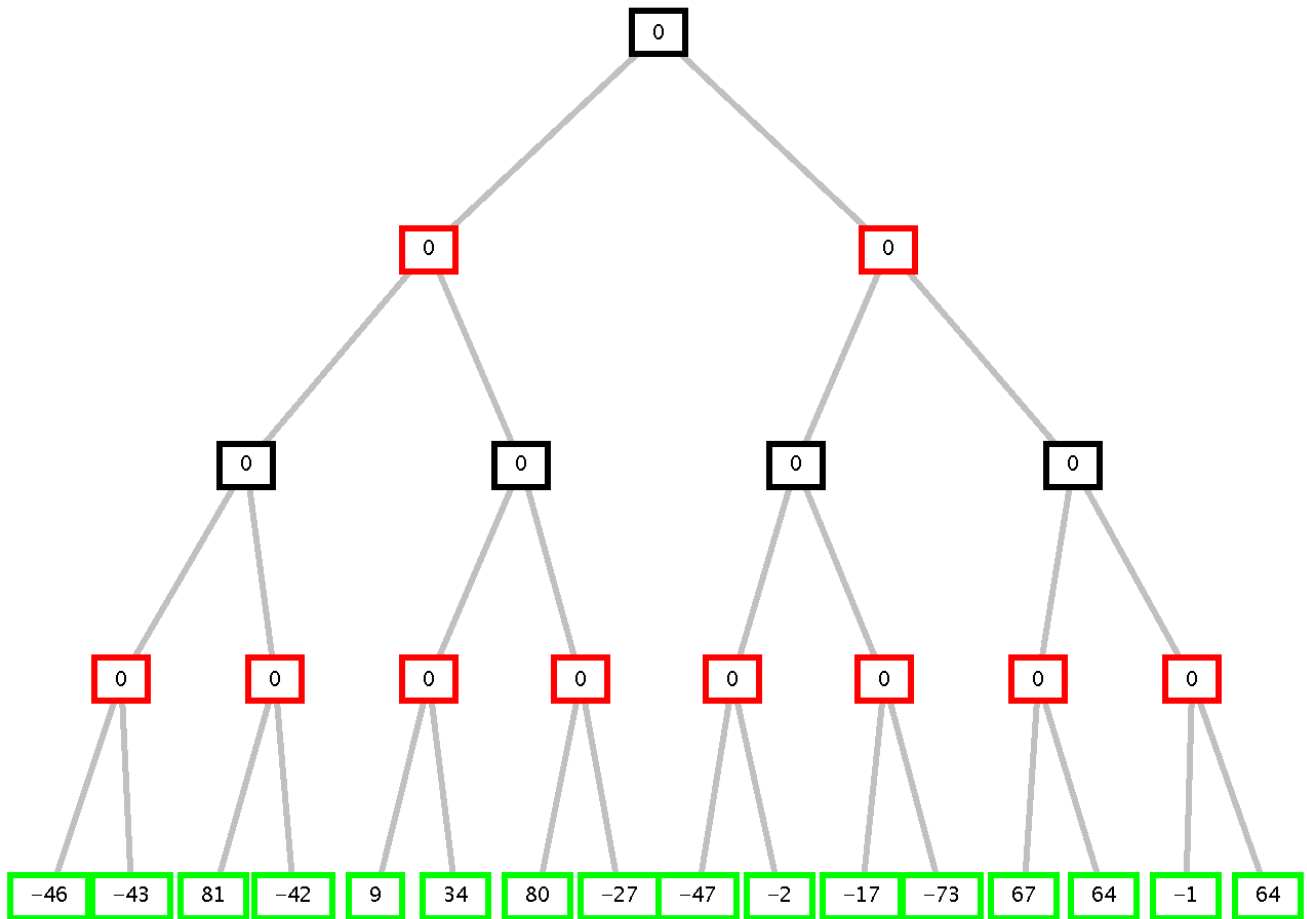
$$\begin{array}{ll} X1::[0..2] & X1*10+X2 \leq 29 \\ X2::[0..9] & X1*10+X2 \geq 1 \\ X3::[1..10] & X1 = (1+X3) \bmod 10 \\ & X2 = (1+X3) \text{ div } 10 \end{array}$$

Si applichi l'Arc-consistenza al CSP, controllando i vincoli per ciascun arco, fino alla quiescenza, e si mostrino i domini finali delle tre variabili.

COGNOME NOME \_\_\_\_\_  
Min-max



Alfa-beta



14 Febbraio 2019 - Soluzioni

**Esercizio 1**

1.  $\forall X (\text{razionale}(X) \rightarrow \text{logico}(X) )$ .
2.  $\forall X (\neg \text{razionale}(X) \rightarrow \neg \text{logico}(X) )$ .
3.  $\exists X \neg \text{logico}(X)$
4.  $\forall X (\neg \text{logico}(X) \rightarrow \text{sbaglia}(X) )$ .
5.  $\text{razionale}(\text{cartesio})$
6.  $\neg \text{razionale}(\text{mirco})$

Query:  $\text{sbaglia}(\text{mirco})$

Goal=QueryNeg:  $\neg \text{sbaglia}(\text{mirco})$

Trasformazione in clausole

- C1.  $\neg \text{razionale}(Y) \vee \text{logico}(Y)$
- C2.  $\text{razionale}(Y) \vee \neg \text{logico}(Y)$
- C3.  $\neg \text{logico}(c)$  // skolem constant
- C4.  $\text{logico}(Y) \vee \text{sbaglia}(Y)$
- C5.  $\text{razionale}(\text{cartesio})$
- C6.  $\neg \text{razionale}(\text{mirco})$
- C7.  $\neg \text{sbaglia}(\text{mirco})$

Risoluzione:

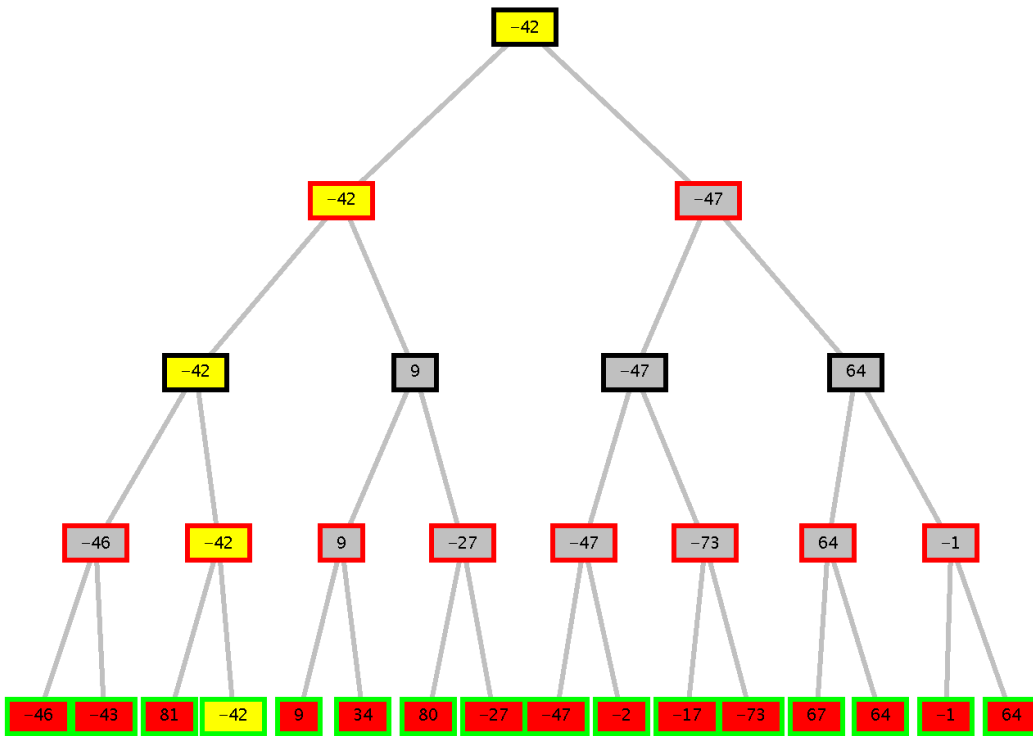
C8 = C7+C4:  $\text{logico}(\text{mirco})$

C9 = C8+C2:  $\text{razionale}(\text{mirco})$

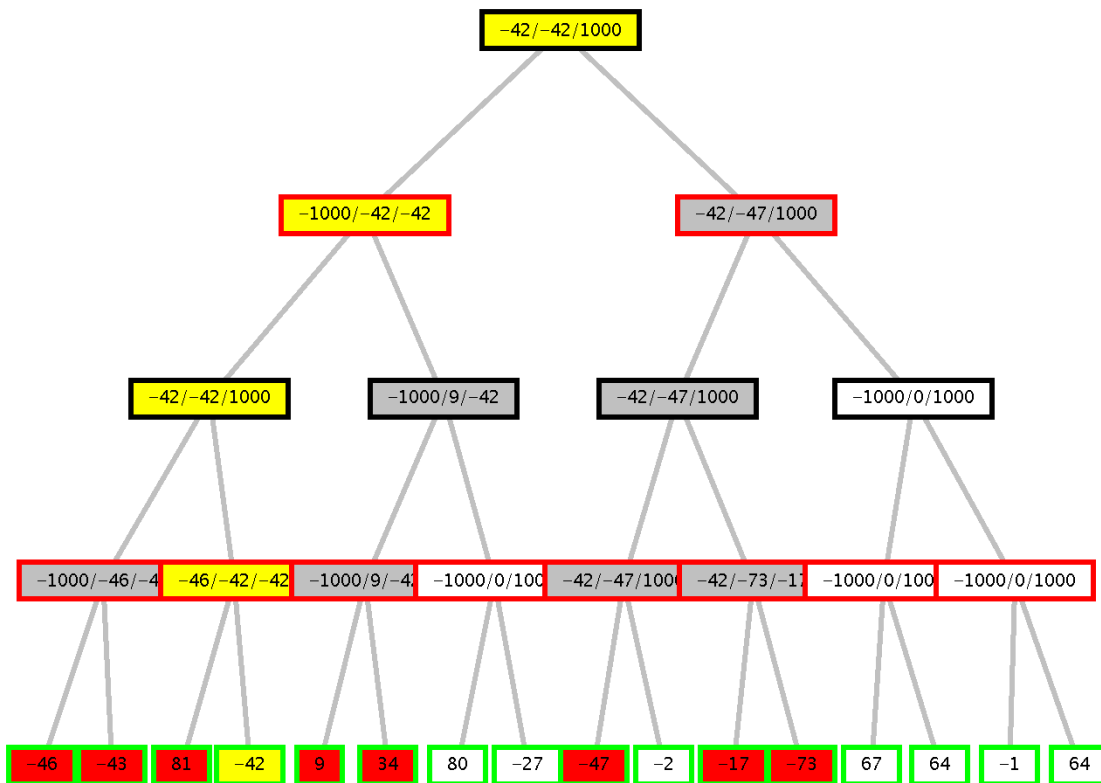
C10 = C9+C6: clausola vuota

**Esercizio 2**

Min-Max:

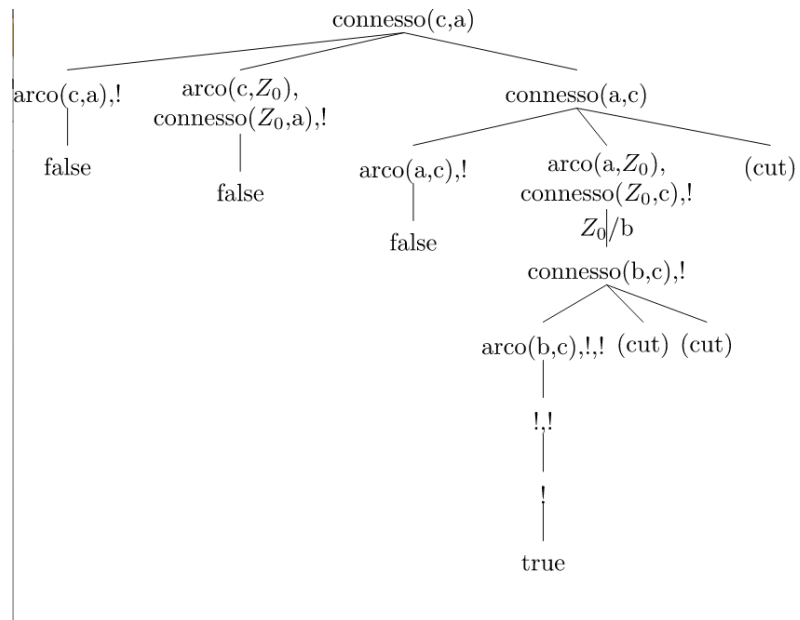


Tagli alfa-beta:



### Esercizio 3

<http://cplint.eu/p/mGWdAOCw.swinb>



#### Esercizio 4

stessaSomma([L], V) :- !, sumlist(L,V).

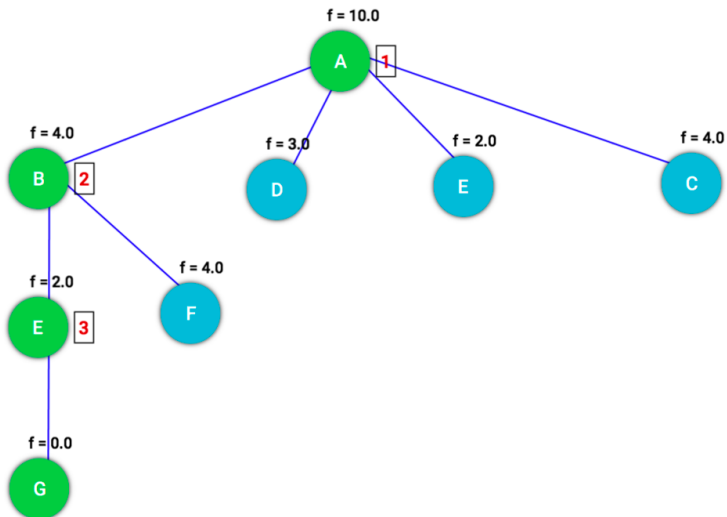
stessaSomma([L|R], V) :- sumlist(L,V), stessaSomma(R,V).

sumlist([H],H) :- !.

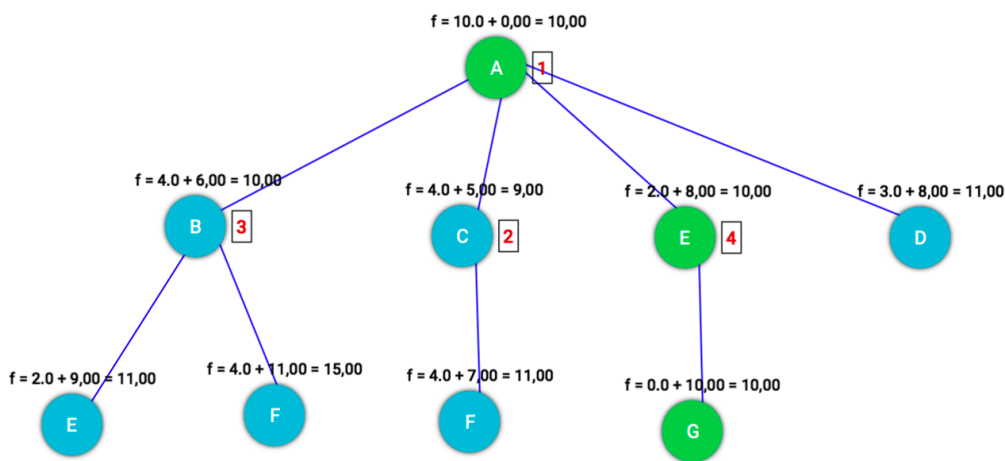
sumlist([X|R],V) :- sumlist(R,Z), V is X+Z.

### Esercizio 5

Depth-first (costo di cammino: 11)



A\* (costo di cammino: 10)



La soluzione trovata da A\* è migliore (ottima).

### Esercizio 6

Applicando l'Arc-consistenza al CSP ottenuto, controllando i vincoli per ciascun arco, nei due sensi, si riduce il dominio delle variabili come segue:

X1::[0..2]

X2::[0,1]

X3::[1,9,10]

Di seguito i passaggi:

Domini:

X1::[0..2]

X2::[0..9]

X3::[1..10]

Vincoli:

a)  $X1 * 10 + X2 \leq 29$

b)  $X1 * 10 + X2 \geq 1$

c)  $X1 = (1 + X3) \bmod 10$

d)  $X2 = (1 + X3) \text{ div } 10$

Prima iterazione

-----

Applico (a).

Per ogni X1, esiste un X2? Si, domini untouched.

Per ogni X2, esiste X1? Si, domini untouched.

Applico (b)

Per ogni X1, esiste un X2? Si, domini untouched.

Per ogni X2, esiste X1? Si, domini untouched.

Applico (c).

Per ogni X1, esiste un X3? Si, domini untouched.

Per ogni X3, esiste un X1? Gli unici valori compatibili di X3 coi domini

esistenti ti X1 sono: 1, 9, 10. Nuovi domini:

X1::[0..2]

X2::[0..9]

X3::[1,9,10]

Applico (d).

Per ogni X2, esiste un X3? Gli unici valori compatibili di X2 sono: 0,

1. Nuovi domini:

X1::[0..2]

X2::[0,1]

X3::[1,9,10]

Per ogni X3, esiste un X2? Si, domini untouched.

Seconda iterazione

-----  
Applico (a):  $X1*10+X2 \leq 29$

Per ogni X1, esiste un X2? Si, domini untouched.

Per ogni X2, esiste X1? Si, domini untouched.

Applico (b):  $X1*10+X2 \geq 1$

Per ogni X1, esiste un X2? Si, domini untouched.

Per ogni X2, esiste X1? Si, domini untouched.

Applico (c):  $X1 = (1+X3) \bmod 10$

Per ogni X1, esiste un X3? Si, domini untouched.

Per ogni X3, esiste un X1? Si, domini untouched.

Applico (d):  $X2 = (1+X3) \div 10$

Per ogni X2, esiste un X3? Si, domini untouched.

Per ogni X3, esiste un X1? Si, domini untouched.

Quiescenza. I domini quindi sono:

X1::[0..2]

X2::[0,1]

X3::[1,9,10]