

FONDAMENTI DI INTELLIGENZA ARTIFICIALE – 1° parte (6 CFU)

10 gennaio 2014 – Tempo a disposizione: 2 h – Risultato: 32/32 punti

Esercizio 1 (7 punti)

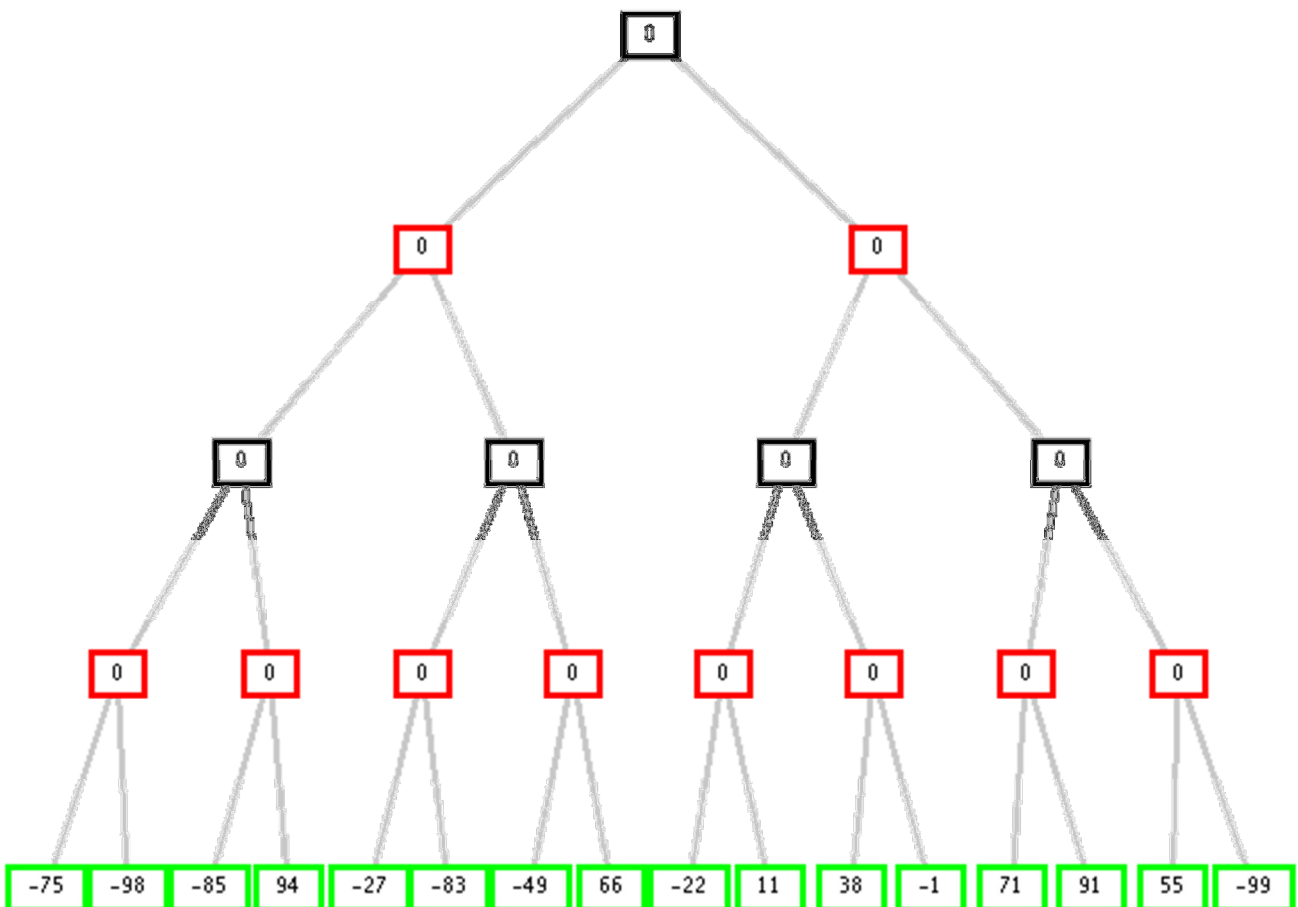
Siano dati i predicati unario evento(X) e binario causa(X, Y) (significato X causa l'effetto Y).
Si formalizzino in *logica dei predicati del I ordine*, le seguenti frasi:

- Ogni evento è causato da un altro evento.
- Dati tre eventi X, Y, Z se X causa Y e Y causa Z allora X causa Z .
- Un evento non può causare sé stesso.
- a, b, c sono eventi
- a causa b
- b causa c

Infine si trasformino in *forma normale congiuntiva* e si dimostri tramite *risoluzione* che a causa c .

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 gli algoritmi *min-max* e *alfa-beta* risolvono il problema.



Esercizio 3 (5 punti)

Si consideri il seguente programma *Prolog*:

```
length([],0):-!.
length(_|T,N):-length(T,N1), N is N1+1.
member(X,[X|_]).
member(X,_|T):-member(X,T).
```

Si mostri l'*albero SLD* relativo al goal:

```
?- length(L,2), member(0,L).
```

Indicare se si ottiene qualche risposta calcolata per la variabile L del goal.

Esercizio 4 (5 punti)

Si scriva un programma *Prolog* `compress(L1,L2)` che data la lista di interi L1 generi in uscita una lista L2 in cui gli elementi consecutivi ripetuti sono stati eliminati e compaiono in singola copia.

Esempi:

```
?- compress([1,2,2,3,3,3,4],L).
yes L=[1,2,3,4];
```

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 tutte di età diversa e (espressa in anni) compresa fra 1 e 10 anni.

Esse possiedono 4 cagnolini, rispettivamente Rococò, Sibilo, Tocco e Uguccione, anch'essi di età diversa e (espressa in anni) compresa fra 1 e 5 anni.

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

Alice ha l'età del cane di Barbara.

Sapendo che l'età delle bambine è per forza un numero pari, così come quella del cane Sibilo, che è il cane di Barbara, ridurre di conseguenza i domini delle variabili.

Si cerchi poi una prima soluzione del problema, e nel corso della ricerca, si applichi il *Forward Checking*. Si scelga come prima variabile da istanziare quella associata al cane Sibilo, poi quella associata ad Alice e poi Barbara, e le successive variabili siano scelte in ordine alfabetico.

Si istanzi ciascuna variabile sempre con il valore più piccolo presente nel dominio, e come detto, dopo ogni istanziazione si applichi il *Forward Checking*.

Esercizio 6 (4 punti)

Si introducano le definizioni di correttezza e completezza di un sistema assiomatico deduttivo.

VOTO:

- Esame da 6 CFU, il voto è determinato da questa I parte
- Esame da 9 CFU, è la media pesata della I parte (che vale 2/3) e della II (che vale 1/3) ovvero il voto finale è dato da: $\frac{2 \times \text{voto I parte} + \text{voto II parte}}{3}$ e varia quindi da 0 ad un massimo di 32 (equivalente alla lode).

FONDAMENTI DI INTELLIGENZA ARTIFICIALE – 2° parte (3 CFU)

10 Gennaio 2014 – Tempo: 45 min – Risultato: 32/32 punti

Esercizio 7 (8 punti)

Si descrivano brevemente gli operatori di “concept forming” [ALL r d], [EXISTS n r], [FILLS r c], [AND d1...dn] introdotti a lezione. Quali di questi sono supportati nella logica descrittiva ALC (*Attributive Language with Complements*)?

Esercizio 8 (12 punti)

Dato un linguaggio proposizionale, con regole “di produzione” del tipo:

`imply([a,b],d). % a,b -> d`

`imply([a,b],e). % a,b -> e`

`imply([a],f). % a -> f`

`imply([a,f],g). % a,f -> g`

si costruisca un meta-interprete che funziona in modalità forward e che - data una lista di fatti in ingresso - sia in grado di derivarne tutte le conseguenze dirette (ovvero che non implicino un concatenamento di più regole).

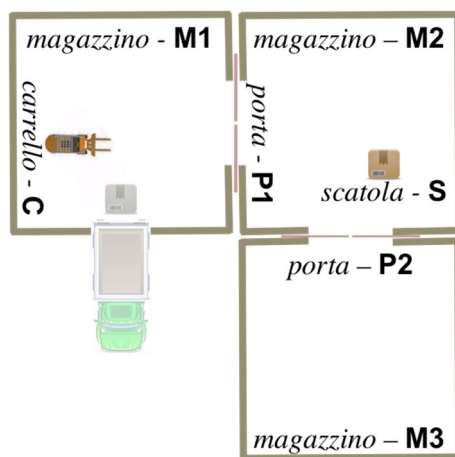
Ad esempio:

`?- infer([a,b],X).`

`yes X=[d,e,f]`

Si supponga che le implicazioni scritte non siano mai cicliche.

Esercizio 9 (12 punti)



Il centro di smistamento di un corriere espresso è formato da 3 magazzini (M1, M2, M3) collegati tra di loro da 2 porte meccanizzate (P1, P2), come schematizzato in figura. Il centro è dotato di un carrello robotico C in grado di spostarsi tra i magazzini per movimentare i pacchi. Inizialmente il carrello C è nella sua posizione di riposo nel magazzino M1 e la scatola S è nel magazzino M2. L'obiettivo è di recuperare la scatola S e di riportarla nel magazzino M1 per la consegna.

Si formalizzi il problema mediante il **linguaggio STRIPS**, avendo cura di definire lo *stato iniziale*, lo *stato finale* e gli operatori

GOTHTROUGH(aDoor, magA, magB) con cui il carrello C attraversa la porta aDoor dal magazzino magA in cui si trova al magazzino magB, e PUSHTHROUGH(aBox, aDoor, magA, magB) con cui il carrello C spinge la scatola aBox attraverso la porta aDoor dal magazzino magA in cui si trova al magazzino magB. A tal proposito si usino i predicati BOX(x) che identifica l'oggetto x come una scatola, INROOM(x, mag) con cui si indica che un oggetto x si trova nel magazzino mag, e CONNECTS(aDoor, magA, magB) con cui si indica che la porta aDoor connette i magazzini magA ed magB.

VOTO:

- Esame da 3 CFU, il voto è determinato da questa II parte
- Esame da 9 CFU, è la media pesata della I parte (che vale 2/3) e della II (che vale 1/3) ovvero il voto finale è dato da: $\frac{2 \times \text{voto I parte} + \text{voto II parte}}{3}$ e varia quindi da 0 ad un massimo di 32 (equivalente alla lode).

FONDAMENTI DI INTELLIGENZA ARTIFICIALE – 1° parte

10 Gennaio 2014 – Soluzioni

Esercizio 1

- Ogni evento è causato da un altro evento.

$\forall X \exists Y: \text{evento}(X) \wedge \text{evento}(Y) \Rightarrow \text{causa}(Y,X)$

(oppure: $\forall X \text{evento}(X) \Rightarrow \exists Y [\text{evento}(Y) \wedge \text{causa}(Y,X)]$)

- Dati tre eventi X,Y,Z se X causa Y e Y causa Z allora X causa Z.

$\forall X \forall Y \forall Z: \text{evento}(X) \wedge \text{evento}(Y) \wedge \text{evento}(Z) \wedge \text{causa}(X,Y) \wedge \text{causa}(Y,Z) \Rightarrow \text{causa}(X,Z)$

- Un evento non può causare sè stesso.

$\forall X \text{evento}(X) \Rightarrow \neg \text{causa}(X,X)$

- a, b, c sono eventi

$\text{evento}(a) \wedge \text{evento}(b) \wedge \text{evento}(c)$

- a causa b

$\text{causa}(a,b)$

- b causa c

$\text{causa}(b,c)$

- Goal: a causa c

$\text{causa}(a,c)$

Forma normale congiuntiva

1. $\neg \text{evento}(X) \vee \neg \text{evento}(f(X)) \vee \text{causa}(f(X),X)$

2. $\neg \text{evento}(X) \vee \neg \text{evento}(Y) \vee \neg \text{evento}(Z) \vee \neg \text{causa}(X,Y) \vee \neg \text{causa}(Y,Z) \vee \text{causa}(X,Z)$

3. $\neg \text{evento}(X) \vee \neg \text{causa}(X,X)$

4. $\text{evento}(a)$

5. $\text{evento}(b)$

6. $\text{evento}(c)$

7. $\text{causa}(a,b)$

8. $\text{causa}(b,c)$

9. $\neg \text{causa}(a,c)$ (goal)

Risoluzione

In più passi, da 2, 4, 5 e 6 derivo:

10. $\neg \text{causa}(a,b) \vee \neg \text{causa}(b,c) \vee \text{causa}(a,c)$

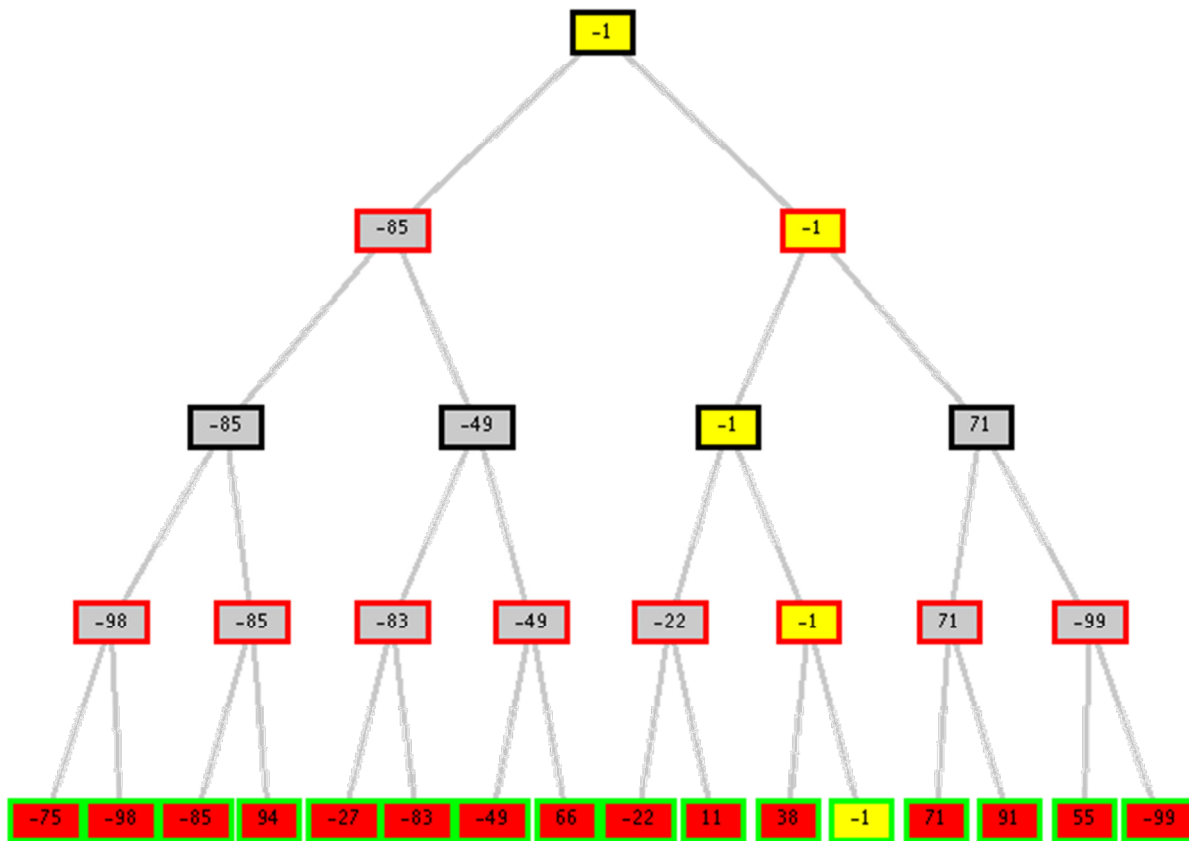
Da 10 e 7 e 8, in due passi, derivo:

11. $\text{causa}(a,c)$

Da 11 e 9 derivo la clausola vuota.

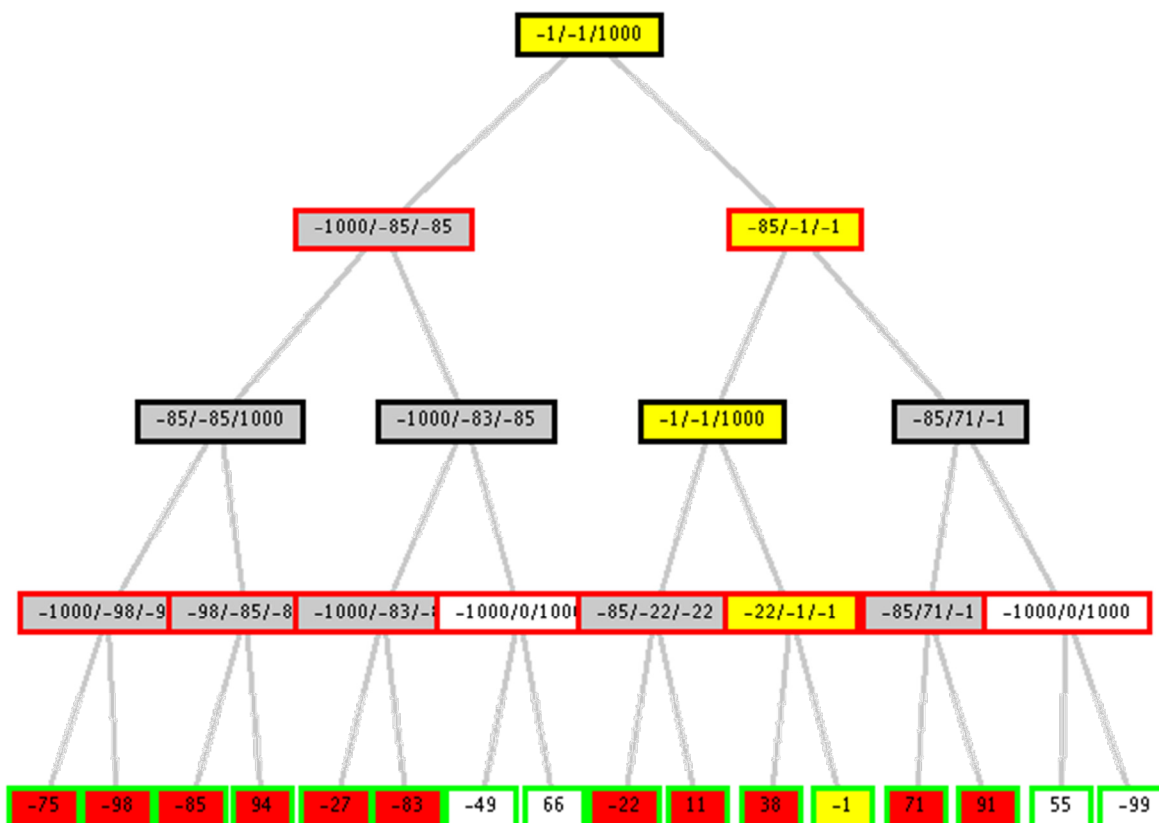
Esercizio 2

min-max:



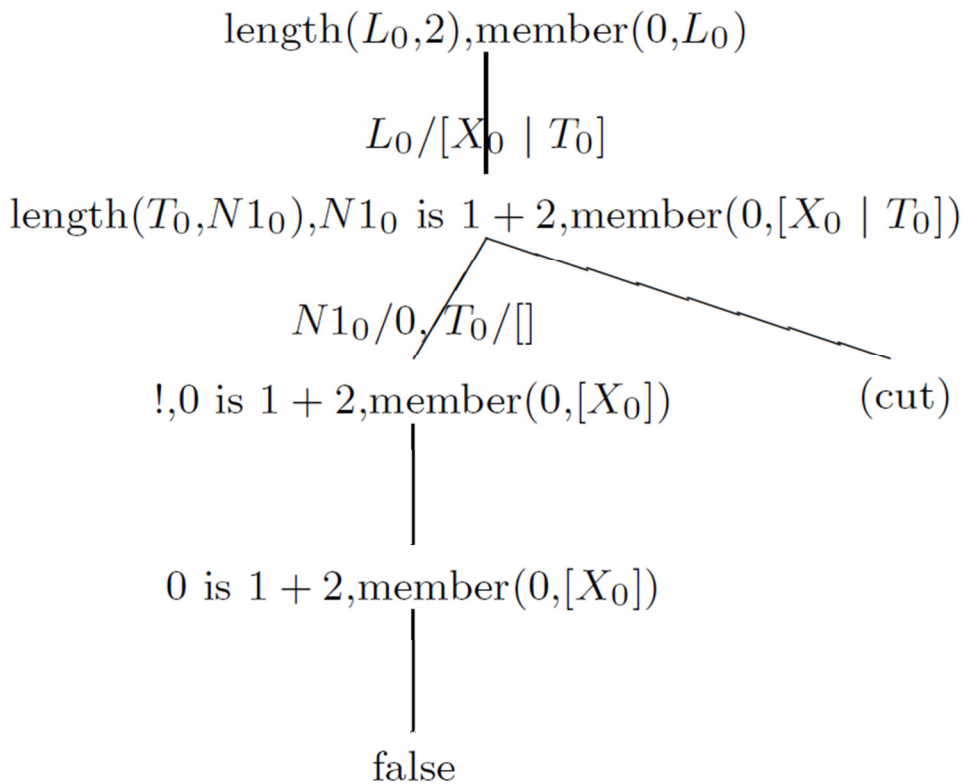
alfa-beta:

I nodi in bianco sono quelli tagliati dall' algoritmo alfa-beta.



Esercizio 3

Albero SLD:



Esercizio 4

```
% Eliminate consecutive duplicates of list elements.  
% compress(L1,L2) :- the list L2 is obtained from the list L1 by  
%   compressing repeated occurrences of elements into a single copy  
%   of the element.  
%   (list,list) (+,?)
```

```
compress([],[]).  
compress([X],[X]).  
compress([X,X|Xs],Zs) :- compress([X|Xs],Zs).  
compress([X,Y|Ys],[X|Zs]) :- X \= Y, compress([Y|Ys],Zs).
```

Esercizio 5

Variabili e domini: una variabile per ogni bambina, e una per ogni cane (iniziale coincidente con nome della variabile)

$$A, B, C, D :: [1..10]$$

$$S, R, S, T, U :: [1..5]$$

Vincoli:

$$A \neq B, B \neq C, C \neq D, R \neq S, S \neq T, T \neq U.$$

$$A = 2 * R$$

$$B = 2 * S$$

$$C = 2 * T$$

$$D = 2 * U$$

$$A = S$$

Sapendo che le età delle bambine sono pari, il dominio è ridotto ai soli pari:

$$A, B, C, D :: [2, 4, 6, 8, 10]$$

e sapendo che Sibilo ha l'età di Alice, anche il suo dominio è ridotto ai soli pari:

$$S :: [2, 4]$$

e quindi anche quello di Alice:

$$A :: [2, 4]$$

Labeling e FC:

	A :: [2, 4, 6, 8, 10]	B :: [2, 4, 6, 8, 10]	C :: [2, 4, 6, 8, 10]	D :: [2, 4, 6, 8, 10]	R :: [1..5]	S :: [2, 4]	T :: [1..5]	U :: [1..5]
S=2	A=2	B=4	C :: [6, 8, 10]	D :: [6, 8, 10]	R=1		T :: [3, 4, 5]	U :: [3, 4, 5]
C=6				D :: [8, 10]			T=3	U :: [4, 5]
D=8								U=4

La prima soluzione è:

$$A=2, B=4, C=6, D=8, R=1, S=2, T=3, U=4$$

Esercizio 6

Vedi slides del corso.

FONDAMENTI DI INTELLIGENZA ARTIFICIALE – 2° parte

10 gennaio 2014 – Soluzioni

Esercizio 7

Vedi slides del corso.

Esercizio 8

```
infer(L,T):-findall(H, (imply(B,H),subset(B,L)), T).
```

```
subset([],_).
```

```
subset([H|T],L):-member(X,L),  
                subset(T,L).
```

Esercizio 9

Stato iniziale:

```
CONNECTS(P1,M1,M2), CONNECTS(P1,M2,M1), CONNECTS(P2,M2,M3), CONNECTS(P2,M3,M2),  
INROOM(C,M1), BOX(S), INROOM(S,M2).
```

Stato finale:

```
CONNECTS(P1,M1,M2), CONNECTS(P1,M2,M1), CONNECTS(P2,M2,M3), CONNECTS(P2,M3,M2),  
INROOM(C,M1), BOX(S), INROOM(S,M1).
```

Operatori:

```
GOTHTROUGH(aDoor, magA, magB)
```

Preconditions:

```
INROOM(C, magA)  $\wedge$  CONNECTS(aDoor, magA, magB)
```

Delete list:

```
INROOM(C, magA)
```

Add list:

```
INROOM(C, magB)
```

```
PUSHTHROUGH(aBox, aDoor, magA, magB)
```

Preconditions:

```
BOX(aBox)  $\wedge$  INROOM(aBox, magA)  $\wedge$  INROOM(C, magA)  $\wedge$  CONNECTS(aDoor, magA, magB)
```

Delete list:

```
INROOM(aBox, magA), INROOM(C, magA)
```

Add list:

```
INROOM(aBox, magB), INROOM(C, magB)
```