

## COMPITO DI APPLICAZIONI DI INTELLIGENZA ARTIFICIALE

13 Dicembre 2007 (Punteggio su 30/30; Tempo 2h )

### Esercizio 1 (punti 8)

Dato il seguente training set S:

Link entranti	Link uscenti	Rilevante
2	2	Si
1	3	No
1	3	No
3	3	No
1	2	Si
2	3	Si
3	3	No
2	2	Si
3	2	No
3	3	No
1	3	Si
3	2	No
1	3	No
2	3	Si
?	2	No
2	3	Si

- Si calcoli l'entropia del training set rispetto all'attributo Rilevante
- Si calcoli il rapporto di guadagno dei due attributi rispetto a questi esempi di training
- si costruisca un albero decisionale ad un solo livello per il training set dato, indicando le etichette delle foglie (numero di esempi finiti nella foglia/numero di esempi finiti nella foglia non appartenenti alla classe della foglia).
- si classifichi l'istanza:

?	3
---	---

### Esercizio 2 (punti 8)

Si devono definire gli orari delle lezioni per 10 docenti su 6 giorni. In ogni giorno gli orari possibili di inizio lezioni sono 10, 12, 14, 16 e le lezioni durano 2 ore.

- La lezione delle 10 prevede un numero di studenti tale da richiedere la presenza di 3 docenti mentre le altre lezioni prevedono un numero di studenti tale da richiedere la presenza di un solo docente.
- Ogni docente non può tenere più di 2 lezioni al giorno.
- Se un docente tiene una lezione non può tenere anche la successiva nello stesso giorno.
- Il docente 4 lavora part time solo i primi tre giorni della settimana.
- Tutti i docenti devono tenere almeno 6 ore di lezioni alla settimana.

Si modelli il problema a vincoli usando possibilmente vincoli globali. Si identifichino alcune simmetrie del problema e si delinei un metodo per eliminarle.

### Esercizio 3 (punti 8)

Si consideri il seguente problema: un carrello ('c') è nella posizione '2' con dei sacchi di cemento ('s') caricati. In posizione '3' si trovano dei mattoni ('m'). La posizione '2' e la posizione '3' sono connesse ad '1'.

Il goal che si vuole raggiungere è che il carrello sia in '1' e mattoni e cemento scaricati in posizione '1', avendo a disposizione le seguenti azioni

Scaricamento materiale

**unload(C,M)**

**PREC:** on(C,M), in(C,X)

**EFFECT:** ¬on(C,M), in(M,X)

Caricamento materiale

**load(C,M)**

**PREC:** in(C,X), in(M,X)

**EFFECT:** on(C,M), ¬in(M,X)

Spostamento del carrello tra due posizioni connesse

**move(C,Loc1,Loc2)**

**PREC:** in(C,Loc1), connected(Loc1,Loc2)

**EFFECT:** in(C,Loc2), ¬in(C,Loc1),

Stato iniziale:

**in(c,2), connected(1,2), connected(2,1), connected(1,3), connected(3,1),  
in(s,2), on(c,s), in(m,3)**

Stato goal: **in(c,1), in(s,1), in(m,1)**

Si mostrino i passi compiuti dall'algorithm POP per risolvere il problema.

### Esercizio 4

- Come viene rappresentato lo stato iniziale, il goal e l'azione load nella pianificazione deduttiva usando la rappresentazione di Kowalsky..
- Si calcoli la least general generalization delle seguenti clausole:  
 $C1 = \{ \text{grade}(a,25), \sim \text{friend}(a,b), \sim \text{friend}(a,c), \sim \text{teacher}(a,t(X,b)), \sim \text{grade}(b,20) \}$ .  
 $C2 = \{ \text{grade}(b,25), \sim \text{friend}(b,c), \sim \text{friend}(a,c), \sim \text{teacher}(a,t(a,Y)), \sim \text{grade}(c,20) \}$ .
- Si descriva cosa è la propagazione dei vincoli e la differenza tra vincoli aritmetici e globali.
- Cosa è una rete neurale multistrato e in quali applicazioni può essere utilizzata.
- Cosa è il reticolo di teta sussunzione.

## SOLUZIONE

### Esercizio 1:

a)  $\text{info}(S) = -8/17 * \log_2 8/17 - 9/17 * \log_2 9/17 = 0.998$  8 #pos, 9 #neg, 17 #tot)

b)

Per calcolare il guadagno dell'attributo Distanza non si usa l'entropia calcolata su tutto il training set ma solo sugli esempi che hanno Distanza noto (insieme F):

$$\text{info}(F) = -8/16 * \log_2 8/16 - 8/16 * \log_2 8/16 = 1$$

$$\text{info}_{\text{Distanza}}(F) = 5/16 * (-4/5 * \log_2 4/5 - 1/5 * \log_2 1/5) + 5/16 * (-3/5 * \log_2 3/5 - 2/5 * \log_2 2/5) + 6/16 * (-1/6 * \log_2 1/6 - 5/6 * \log_2 5/6) = // \text{tre entropie calcolate sui tre sottoinsiemi con diverso valore dell'attr. Distanza} \\ = 0.312 * 0.722 + 0.312 * 0.971 + 0.375 * 0.650 = 0.772$$

$$\text{gain}(\text{Distanza}) = 16/17 * (1 - 0.772) = 0.215$$

Entropia del partizionamento:

$$\text{splitinfo}(\text{Distanza}) = -5/17 * \log_2(5/17) - 5/17 * \log_2(5/17) - 6/17 * \log_2(6/17) - 1/17 * \log_2(1/17) = 1.809$$

// calcolato sui tre sottoinsiemi con Diverso valore Distanza e il sottoinsieme con Distanza non noto

$$\text{gainratio}(\text{Distanza}) = 0.215 / 1.809 = 0.119$$

Per calcolare il guadagno dell'attributo Uguali non si usa l'entropia calcolata su tutto il training set ma solo sugli esempi che hanno Uguali noto (insieme F):

$$\text{info}(F) = -7/15 * \log_2 7/15 - 8/15 * \log_2 8/15 = 0.997$$

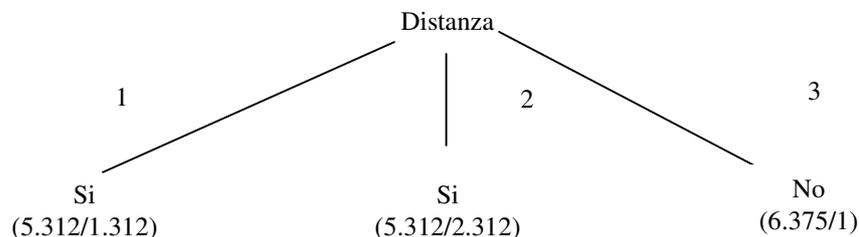
$$\text{info}_{\text{Uguali}}(F) = 8/15 * (-4/8 * \log_2 4/8 - 4/8 * \log_2 4/8) + 7/15 * (-3/7 * \log_2 3/7 - 4/7 * \log_2 4/7) = \\ = 0.533 * 1 + 0.467 * 0.985 = 0.993$$

$$\text{gain}(\text{Uguali}) = 15/17 * (0.997 - 0.993) = 0.004$$

$$\text{splitinfo}(\text{Uguali}) = -7/17 * \log_2(7/17) - 8/17 * \log_2(8/17) - 2/17 * \log_2(2/17) = 1.402$$

$$\text{gainratio}(\text{Uguali}) = 0.004 / 1.402 = 0.003$$

c) L'attributo scelto per la radice dell'albero è Distanza (maggiore gainratio).



d) l'istanza viene divisa in tre parti, di peso rispettivamente  $5.312/17=0.312$ ,  $5.312/17=0.312$  e  $6.375/17=0.375$ . La prima parte viene mandata lungo il ramo 1 e viene classificata come Si con probabilità  $4/5.312=75.3\%$  e come No con probabilità  $1-75.3\%=24.7\%$ . La seconda parte viene mandata lungo il ramo 2 e viene classificata come Si con probabilità  $3/5.312=56.5\%$  e come No con probabilità  $1-56.5\%=43.5\%$ . La terza parte viene mandata lungo il ramo 3 e viene classificata come No con probabilità  $5.375/6.375=84.3\%$  e come Si con probabilità  $1-84.3\%=15.7\%$ . Quindi in totale la classificazione dell'istanza è

$$\text{Si: } 0.312 * 75.3\% + 0.312 * 56.5\% + 0.375 * 15.7\% = 47.0\%$$

$$\text{No: } 0.312 * 24.7\% + 0.312 * 43.5\% + 0.375 * 84.3\% = 52.9\%$$

## ESERCIZIO 2

Dieci<sub>i1</sub> primo docente della lezione delle dieci del giorno i per i = 1..6

Dieci<sub>i2</sub> secondo docente della lezione delle dieci del giorno i per i = 1..6

Dieci<sub>i3</sub> terzo docente della lezione delle dieci del giorno i per i = 1..6

Dodici<sub>i</sub> docente della lezione delle dodici del giorno i per i = 1..6

Quattordici<sub>i</sub> docente della lezione delle quattordici del giorno i per i = 1..6

Sedici<sub>i</sub> docente della lezione delle sedici del giorno i per i = 1..6

Inizialmente i domini sono [1..10] e rappresentano i docenti

alldifferent([Dieci<sub>i1</sub>, Dieci<sub>i2</sub>, Dieci<sub>i3</sub>]) per ogni i

Ogni docente non può tenere più di due lezioni nello stesso giorno

gcc([Dieci<sub>i1</sub>, Dieci<sub>i2</sub>, Dieci<sub>i3</sub>, Dodici<sub>i</sub>, Quattordici<sub>i</sub>, Sedici<sub>i</sub>], j, 0, 2) per ogni i = 1..6 e per ogni j = 1..10

Se un docente tiene una lezione non può tenere anche la successiva nello stesso giorno.

Dieci<sub>ij</sub> ≠ Dodici<sub>i</sub> per ogni i=1..6 e per j =1..3

Dodici<sub>i</sub> ≠ Quattordici<sub>i</sub> per ogni i=1..6

Sedici<sub>i</sub> ≠ Quattordici<sub>i</sub> per ogni i=1..6

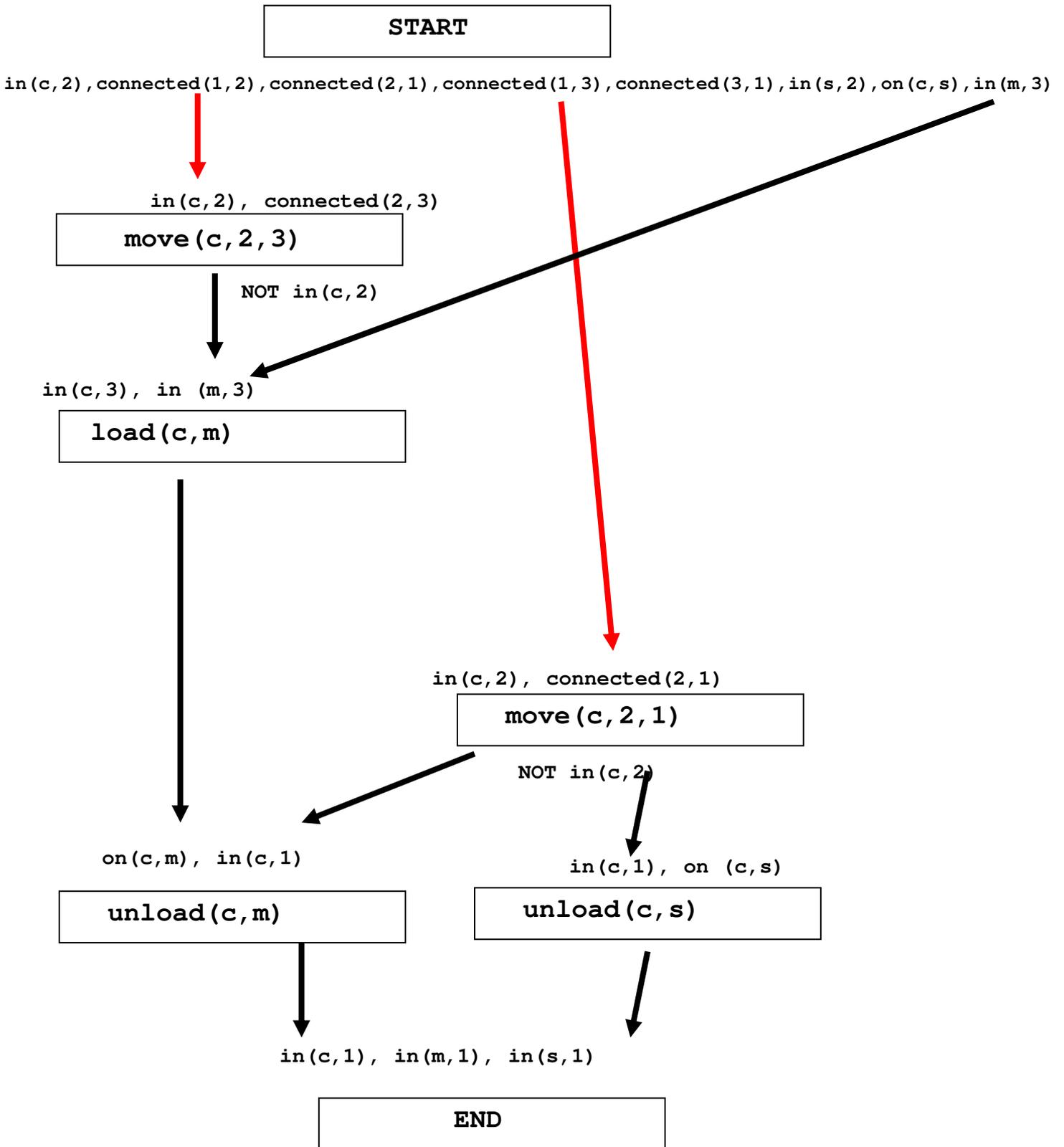
Il docente 4 lavora part time solo i primi tre giorni della settimana.

Dieci<sub>i1</sub> ≠ 4, Dieci<sub>i2</sub> ≠ 4, Dieci<sub>i3</sub> ≠ 4, Dodici<sub>i</sub> ≠ 4, Quattordici<sub>i</sub> ≠ 4, Sedici<sub>i</sub> ≠ 4, per i = 4,5,6

Tutti i docenti devono tenere almeno 6 ore di lezioni alla settimana.

gcc([Dieci<sub>11</sub>, Dieci<sub>12</sub>, Dieci<sub>13</sub>, Dodici<sub>1</sub>, Quattordici<sub>1</sub>, Sedici<sub>1</sub>, Dieci<sub>21</sub>, Dieci<sub>22</sub>, ....., Dieci<sub>61</sub>, Dieci<sub>62</sub>, Dieci<sub>63</sub>, Dodici<sub>6</sub>, Quattordici<sub>6</sub>, Sedici<sub>6</sub>], j, 3, 6) per ogni j = 1..10

Esercizio 3



Questo piano contiene un threat: infatti i due *causal link* in rosso sono minacciato dagli effetti delle due azioni *move* che come effetto contengono *not in(c,2)*. In questo caso non si possono applicare la demotion e la promotion. Serve una azione che ristabilisca *in(c,2)*

**START**

$in(c, 2), connected(1, 2), connected(2, 1), connected(1, 3), connected(3, 1), in(s, 2), on(c, s), in(m, 3)$

$in(c, 2), connected(2, 3)$

**move(c, 2, 3)**

NOT  $in(c, 2)$

$in(c, 3), in(m, 3)$

**load(c, m)**

$in(c, 3), connected(3, 2)$

**move(c, 3, 2)**

$in(c, 2), connected(2, 1)$

**move(c, 2, 1)**

NOT  $in(c, 2)$

$on(c, m), in(c, 1)$

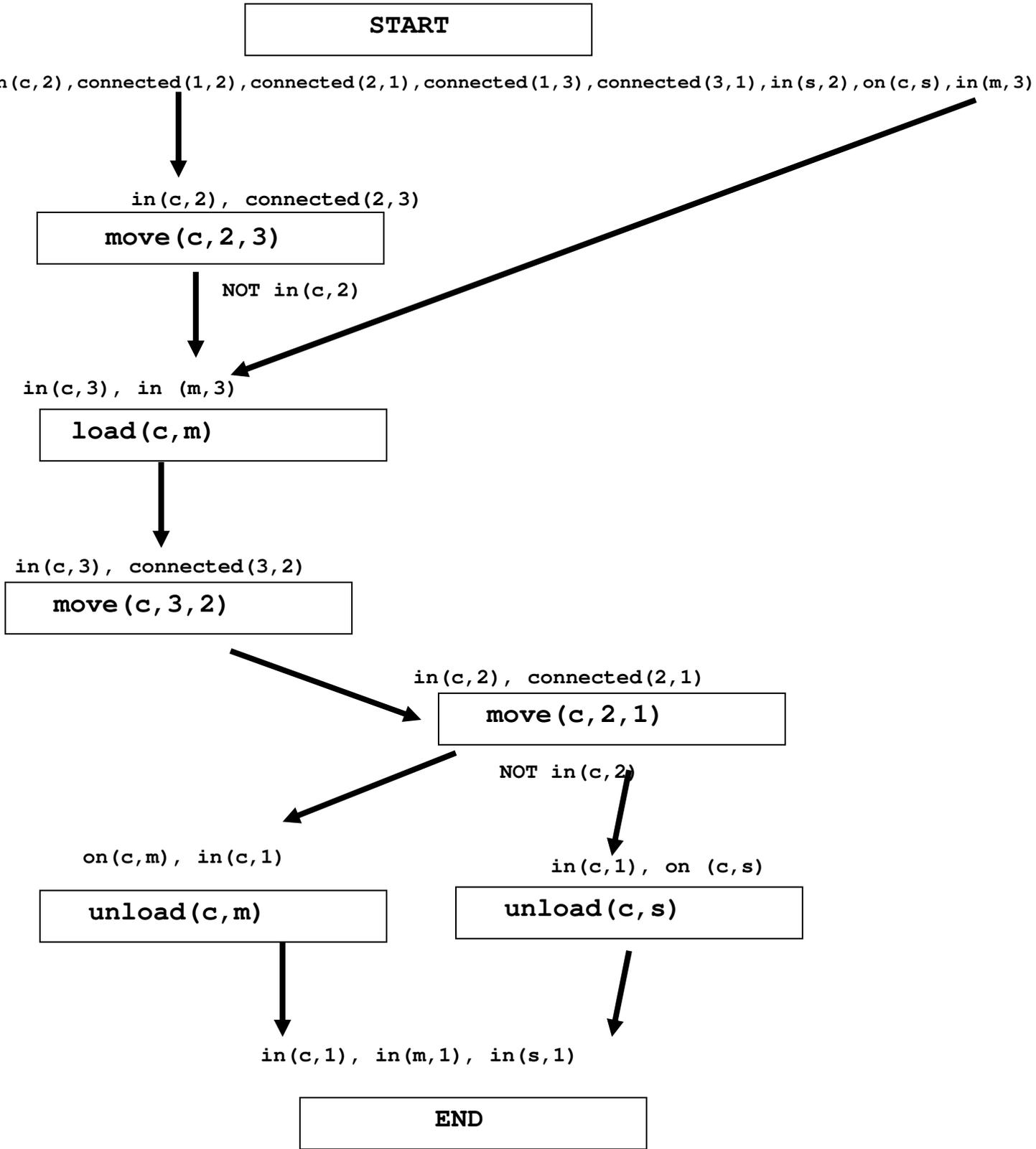
**unload(c, m)**

$in(c, 1), on(c, s)$

**unload(c, s)**

$in(c, 1), in(m, 1), in(s, 1)$

**END**



**F**

**Esercizio 4**

$\text{lgg}(C1,C2) = \{ \text{grade}(A,25), \sim\text{friend}(B,C), \sim\text{friend}(a,C), \sim\text{friend}(A,c), \sim\text{friend}(a,c), \sim\text{teacher}(a,t(D,E)), \sim\text{grade}(C,20) \}$ .