Facoltà di Ingegneria dell'informazione

Ingegneria della conoscenza 2009–10

Appello del 7 luglio 2010 – Soluzioni

1 (4 pt.)

Con riferimento a un sistema logico qualsiasi, spiegare i concetti di *decidibilità* e *semidecidibilità* del reasoning. Fare un esempio di logica con reasoning decidibile e uno di logica con reasoning semidecidibile.

Vedi le dispense, par. 2.2.

2 (10 pt.)

Utilizzando la notazione DL, definire in OWL 2 DL la seguente ontologia concernente le liste lineari di interi. Di ogni proprietà specificare dominio e codominio. Identificare le eventuali specifiche che non possono essere rappresentate in OWL 2 DL, spiegare perché e, se possibile, fornire una rappresentazione approssimata. Ove possibile utilizzare asserzioni di ABox. Specifiche:

1. ogni lista ha al massimo un nodo come testa; liste e nodi sono disgiunti;
2. a ogni nodo sono associati: esattamente un'informazione, che è numero intero; e al massimo un successore, che è un nodo;
3. le liste vuote sono le prive di testa; i nodi iniziali sono i nodi che sono testa di una lista; i nodi terminali sono i nodi privi di successore;
4. un nodo x precede un nodo y se, e solo se, esiste una catena di relazioni di successore che porta da x a y (la catena deve avere lunghezza uguale o superiore a 1);
5. un nodo appartiene a una lista se: è il nodo iniziale della lista; oppure è preceduto da un nodo che appartiene alla lista;
6. la lista `lista1` ha `nodo1` come testa, che ha `nodo2` come successore; l'informazione di `nodo1` è l'intero `25`, quella di `nodo2` è l'intero `-7`.

Dire se dalla vostra ontologia è deducibile quanto segue (formulare le interrogazioni, dire quale servizio di ragionamento viene invocato, indicare la risposta e giustificarla brevemente):

7. è possibile che un nodo sia contemporaneamente nodo iniziale e nodo terminale;
8. il nodo `nodo2` di `lista1` (domanda 6) è un nodo terminale.

1. haTesta: Lista $\longrightarrow$ Nodo Lista $\sqsubseteq\ \leq 1$ haTesta Lista $\sqcap$ Nodo $\sqsubseteq \bot$
2. haInfo: Nodo $\longrightarrow_D$ integer haSucc: Nodo $\longrightarrow$ Nodo Nodo $\sqsubseteq$ =1 haInfo $\sqcap \leq 1$ haSucc
3. ListaVuota $\equiv$ Lista $\sqcap\ \neg\exists$haTesta NodoIniziale $\equiv \exists$haTesta$^-$ NodoTerminale $\equiv$ Nodo $\sqcap\ \neg\exists$haSucc
4. precede: Nodo $\longrightarrow$ Nodo haSucc $\sqsubseteq$ precede *Tra*(precede)
   Si tratta di un'approssimazione, che rappresenta solo la parte condizione sufficiente e non la condizione necessaria per la precedenza
5. appartieneA: Nodo $\longrightarrow$ Lista haTesta$^-$ $\sqsubseteq$ appartieneA
   precede$^-$ $\circ$ appartieneA $\sqsubseteq$ appartieneA (oppure haSucc$^-$ $\circ$ appartieneA $\sqsubseteq$ appartieneA, equivalente e più efficiente)
6. haTesta(lista1,nodo1) haSucc(nodo1,nodo2) haInfo(nodo1,"25"^^integer) haInfo(nodo2,"–7"^^integer)
7. ?– NodoIniziale $\sqcap$ NodoTerminale (verifica di soddisfacibilità) $\Rightarrow$ true
   La classe è soddisfacibile perché tutti gli assiomi sono verificati da una lista costituita da un solo nodo (che è iniziale e anche terminale)
8. ?– NodoTerminale(nodo1) (instance check) $\Rightarrow$ false
   Ricordiamo che la risposta "false" significa: non deducibile. In effetti dall'ontologia non è deducibile che nodo2 sia privo di successore.

3 (4 pt.)

Per ciascuna delle espressioni DL seguenti, e prestando attenzione alla differenza fra classi ed enunciati:
- tradurre l'espressione in italiano (senza utilizzare variabili o altri termini tecnici della logica o della teoria degli insiemi)
- specificare la semantica dell'espressione in termini di modelli $M = \langle \Delta, -^I \rangle$:

1. CartaDaGioco $\sqcap\ \neg$(haSeme $\ni$ cuori)
   Le carte da gioco non di cuori (classe)
   $(...)^I$ = CartaDaGioco$^I \cap (\Delta \setminus \{x \in \Delta \mid \langle x,\text{cuori}^I\rangle \in \text{haSeme}^I\}$

2. CartaDaGiocoRossa $\equiv$ (haSeme $\ni$ cuori) $\sqcup$ (haSeme $\ni$ quadri)
   Le carte da gioco rosse sono quelle di cuori o di quadri (enunciato di TBox)
   M |= ... sse CartaDaGiocoRossa$^I = \{x \in \Delta \mid \langle x,\text{cuori}^I\rangle \in \text{haSeme}^I\} \cup \{x \in \Delta \mid \langle x,\text{quadri}^I\rangle \in \text{haSeme}^I\}$

3. $\exists$genitoreDi $\sqcap\ \forall$genitoreDi.Uomo (andrea)
   Andrea è genitore di soli figli maschi (asserzione di ABox)
   M |= ... sse andrea$^I \in \{x \in \Delta \mid$ per qualche $y \in \Delta$, $\langle x,y\rangle \in \text{genitoreDi}^I\}$
   $\cap \{x \in \Delta \mid$ se per qualche $y \in \Delta$ si ha $\langle x,y\rangle \in \text{genitoreDi}^I$, allora $y \in \text{Uomo}^I\}$

4. confinaCon $\sqsubseteq$ confinaCon$^-$
   se un oggetto confina con un altro, il secondo confina con il primo (enunciato di RBox)
   M |= ... sse confinaCon$^I \subseteq (\text{confinaCon}^I)^{-1}$