Logica computazionale

Esame scritto, Febbraio 2024

Risultati

Attenzione: i voti sono stati leggermente rivisti dopo le 12:00 del 21 febbraio a causa di un errore di trascrizione nella lista dei voti caricata precedentemente.

Matricola	Voto
227233	30 e lode
222609	30 e lode
227487	30 e lode
230527	29
228055	27
218337	26
226942	26
227818	25
205793	25
218349	22
226623	21
227743	21
209037	20
231822	20
218313	20
231219	19
209313	17
201543	17
234815	17
209929	17
226785	17
227046	16
230158	15
227306	15
227226	15
226746	14
227114	13
203123	6

Soluzioni

TOTALE: 36pt   durata 100 minuti

1. Teoria Rappresentazione mentale (2pt)

Dire quali delle seguenti affermazioni sono vere (una o più):

1. Assumendo che due persone possano avere la stessa rappresentazione mentale analogica, queste persone possono avere due diverse rappresentazioni linguistiche della stessa rappresentazione mentale analogica
2. Ciò che noi percepiamo può essere modellato in termini di insiemi ed elementi di insiemi
3. Una funzione di interpretazione può essere polisemica, ossia assegnare due elementi del dominio allo stesso elemento del linguaggio
4. Una funzione di interpretazione può essere sinonimica, ossia assegnare lo stesso elemento del dominio a due diversi elementi del linguaggio

SOLUZIONE

1. vero
2. vero
3. Falso
4. vero

2. Teoria Logic / funzione di interpretazione /model checking (3pt)

Dire quali delle seguenti affermazioni (una o più) sono vere in Logica delle proposizioni (LOP):

1. Data una Teoria T, M è un modello di T se è una interpretazione che rende vere tutte le formule di T
2. Dato un linguaggio L, l'interpretazione di una congiunzione è uguale alla congiunzione dell'interpretazione dei congiunti. In formule:

$$I\ (P1 \land P2) = I(P1)\ \land\ I(P2)$$

3. $I \vDash P1 \land P2$, se $I \vDash P1$ e $I \vDash P2$, dove I è una interpretazione e P1 e P2 sono proposizioni qualunque.
4. Dato un linguaggio L che permetta il seguente insieme di proposizioni atomiche $\{P\} = \{P1, P2\}$ e la teoria T con zero assiomi, ovvero T={}, allora T ha due modelli

SOLUZIONE

1. Vera per la definizione di modello

2. Falsa, La funzione di interpretazione si applica solo alla formule atomiche
3. Vera, per definizione di entailment
4. Falsa, non avendo assiomi T è vera in tutti i modelli, che sono in effetti 4.

3. Teoria semantica LOD (2pt)

Si considerino le seguenti affermazioni riguardo la semantica della Logica delle Descrizioni (LOD). Da notare che il simbolo “\” indica la differenza tra insiemi, D è il dominio di interpretazione, I è la funzione di interpretazione.

Indicare quali delle seguenti affermazioni sono VERE (una o più):

1. Esistono una o più formule la cui interpretazione è l’insieme vuoto
2. I($\neg A$) = D\ I(A)
3. I(∃R.C) = {d ∈ D | esiste e ∈ D con (d,e) ∈ I(R) ed e ∈ I(C)}
4. I(∀R.C) = {d ∈ D | per tutte le e ∈ D se (d,e) ∈ I(C) allora e ∈ I(R)}

SOLUZIONE

1. Vero
2. Vero
3. Vero
4. Falso

4. Teoria semantica LOP (2pt)

Dire quale delle seguenti affermazioni sotto indicate (una o più) sono VERE riguardo la Logica delle Proposizioni (LOP):

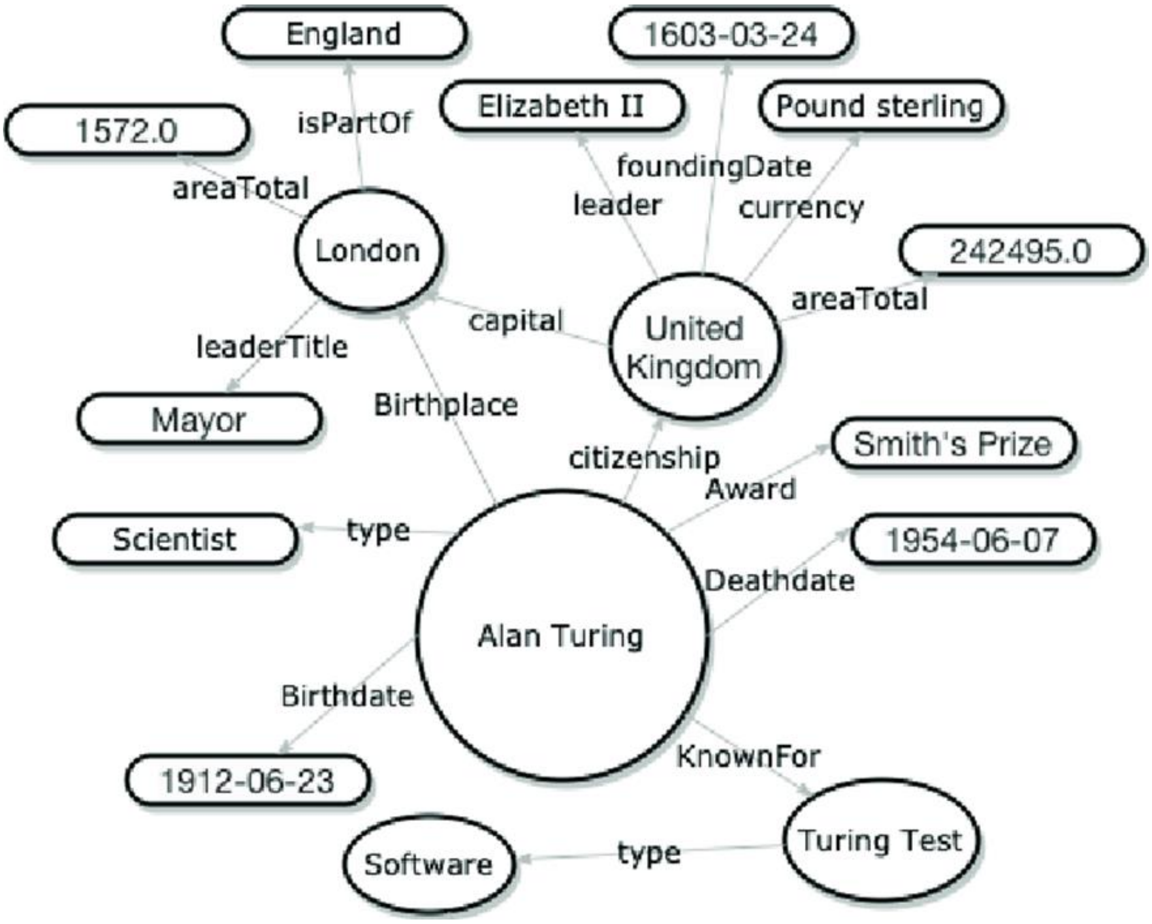
1. $\neg ((A \wedge \neg B) \vee B) \equiv \neg(A \wedge \neg B) \wedge \neg B$ è una applicazione della legge di DeMorgan
2. $(A \supset (B \vee C) \equiv (A \supset B) \wedge (B \supset C)$ è un “basic fact” (una tautologia) di LOP che descrive la relazione fra implicazione e disgiunzione
3. $(A + B) \equiv (\neg A \vee B) \wedge (\neg B \vee A)$ è la definizione del simbolo logico xor (+)

SOLUZIONE

1. Vera, basta prendere C = $(A \wedge \neg B)$ nella legge di De Morgan
2. Falsa. La versione corretta è $(A \supset (B \vee C) \equiv (A \supset B) \vee (A \supset C)$. In effetti non è neanche una tautologia
3. Falsa, vedere la definizione di xor (+)

5. Esercizio I2F da KG a linguaggio LoE (4pt)

Tenendo come riferimento il grafo di conoscenza informale ("Knowledge graph (KG)") riportato in figura, indicare quali delle seguenti affermazioni circa la corrispondenza tra il grafo e la sua formalizzazione nella logica delle entità ("Logic of Entities", LOE) sono VERE (una o più):



NOTAZIONE: l'etichetta "type" degli archi in figura viene usata per asserire l'etype delle entità.

1. **(Falsa)** Il nodo Alan Turing potrebbe essere un elemento di un dtype
2. **(Vera)** Non è possibile determinare l'etype di London se non il fatto che è un'istanza dell'etype più generale, che chiamiamo Entity
3. **(Falsa)** Mayor è necessariamente un etype
4. **(Vera)** L'asserzione Scientist(AlanTuring) può far parte della teoria che formalizza il KG in figura
5. **(Vera)** L'asserzione Birthplace(AlanTuring, London) può far parte della teoria che formalizza il KG in figura
6. **(Falsa)** L'asserzione areaTotal(1572.0, London) può far parte della teoria che corrisponde al grafo
7. **(Falsa)** Gli etype devono includere Software, Scientist e Place
8. **(Vera)** Il nodo Elizabeth II potrebbe essere un elemento di un dtype

SOLUZIONE:

La (1) è falsa perché i dype devono essere nodi foglia.

La (2) è vera perché non viene esplicitato il tipo attraverso gli archi.

La (3) è falsa perché Mayor potrebbe essere rappresentato come un valore di un dtype.

La (4) e la (5) sono vere perché c’è un corrispondente arco nel grafo.

La (6) è falsa perché nel grafo la direzione dell’arco è opposta.

La (7) è falsa perché Place non è rappresentato nel grafo.

La (8) è vera perché è un nodo foglia.

6. Esercizio da NL a LOD (4 punti)

Indicare quali delle seguenti affermazioni circa la corrispondenza tra linguaggio naturale e loro formalizzazione nella logica delle descrizioni (“Logic of Descriptions”, LOD) sono VERE (una o più):

1. **(Falsa)** La formalizzazione di “Le banane sono dei frutti che possono essere di colore giallo o rosso” è

   $\text{Banana} \sqsubseteq \text{Frutto} \sqcap \exists \text{Colore}.(\text{Giallo} \sqcup \text{Rosso})$

2. **(Vera)** La formalizzazione di “I pinguini sono uccelli che non posseggono ali” è

   $\text{Pinguino} \sqsubseteq \text{Uccello} \sqcap \neg \exists \text{possiede}.\text{Ala}$

3. **(Falsa)** La formalizzazione di “Ci sono rinoceronti che non possiedono il corno” è

   $\text{Rinoceronte} \sqsubseteq \neg \exists \text{possiede}.\text{Corno}$

4. **(Falsa)** La formalizzazione di “I quadrupedi sono animali che hanno zampe” è

   $\text{Quadrupede} \sqsubseteq \text{Animale} \sqcap \forall \text{possiede}.\text{Zampa}$

5. **(Vera)** La formalizzazione di “Tutti gli impiegati sono di nazionalità italiana e non hanno una laurea come titolo di studio” è

   $\text{Impiegato} \sqsubseteq \exists \text{Nazionalità}.\text{Italiano} \sqcap \neg \exists \text{TitoloDiStudio}.\text{Laurea}$

SOLUZIONE:
La (2) e la (5) sono banalmente vere.

La (1) è falsa perché la grammatica LOD prevede che il target di un quantificatore debba essere necessariamente un dtype o un etpye; una traduzione corretta è Banana ⊑ Frutto ⊓ (∃Colore.Giallo ⊔ ∃Colore.Rosso).

La (3) è falsa perché andrebbe formalizzata come Rinoceronte ⊓ ¬∃possiede.Corno)

La (4) è falsa perché occorre il quantificatore esistenziale, altrimenti risulta che i quadrupedi sono fatti solo di zampe

7. Esercizio I2F da NL a LOP (3 punti)

Indicare quali delle seguenti traduzioni dall'italiano alla logica delle proposizioni (LOP) è corretta (una o più):

1. La frase "Angelo andrà al mare, considerato che Bruno e Carlo non andranno" (si assuma che "considerato che" è la traduzione dall'inglese di "provided that") può essere tradotta come Angelo ⊃ (¬Bruno ∧ ¬Carlo).

2. La frase "Davide non giocherà e giocherà certamente Carlo; oppure Davide giocherà e Carlo no." può essere tradotta come (DavideGioca + Carlo)

3. La frase "Davide giocherà se gioca Carlo" può essere tradotta come (¬Carlo ∨ Davide)

SOLUZIONE. La 1 è falsa perché va tradotta nel verso opposto dell'implicazione, ovvero come (¬Bruno ∧ ¬Carlo) ⊃ Angelo. La 2 è banalmente vera. La 3 è vera, ed è equivalente a Carlo ⊃ Davide.

8. Esercizio entailment in LODE (5 punti)

Data la seguente TBOX T in linguaggio LODE:

Allevatore ≡ Persona ⊓ ∃alleva.Animale

Pastore ≡ Allevatore ⊓ ∀alleva.Pecora

Casaro ≡ Pastore ⊓ ∃produce.Formaggio

e la seguente ABOX A in linguaggio LODE:

Persona(Antonio)

Casaro(Giuseppe)

Pastore(Davide)

alleva(Giuseppe, Dolly)

produce(Davide, Asiago)

Denotiamo inoltre con EG il LODE Entity Graph che risulta dalla composizione, attraverso lo sviluppo (“unfolding”) della TBOX e espansione (“expansion”) della ABOX.

Indicare quale delle seguenti affermazioni sono vere (una o più):

1. T ⊨ Pastore(Giuseppe)
2. T ⊨ Animale(Dolly)
3. T ⊨ Persona(Davide)
4. L’EG contiene un solo nodo che rappresenta entità anonime
5. L’EG contiene esattamente due nodi che rappresentano entità anonime
6. L’EG contiene più di due nodi che rappresentano entità anonime

SOLUZIONE.

Cominciamo a calcolare l’unfolding della TBOX:

Allevatore ≡ Persona ⊓ ∃alleva.Animale

Pastore ≡ Persona ⊓ ∃alleva.Animale ⊓ ∀alleva.Pecora

Casaro ≡ Persona ⊓ ∃alleva.Animale ⊓ ∀alleva.Pecora ⊓ ∃produce.Formaggio

Calcoliamo quindi l’espansione A’ della ABOX rispetto alla TBOX unfolded:

Persona(Antonio)

Casaro(Giuseppe), alleva(Giuseppe, Dolly), Persona(Giuseppe), Pecora(Dolly), Animale(Dolly), Pastore(Giuseppe), produce(Giuseppe, X1), Formaggio(X1)

Pastore(Davide), Persona(Davide), alleva(Davide, X2), Pecora(X2), Animale(X2)

produce(Davide, Asiago)

Da osservare che essendo Casaro $\sqsubseteq \exists$alleva.Animale $\sqcap \forall$alleva.Pecora gli anonimi generati dalla condizione col quantificatore esistenziale devono necessariamente rispettare anche la condizione col quantificatore universale. Di conseguenza, l'anonimo deve coincidere con Dolly che deve pertanto essere sia Pecora che Animale. Inoltre, essendo Pastore più specifico di Casaro, Giuseppe è anche Pastore. Nel caso di Davide, invece, non avendo istanze di alleva, resta l'anonimo X2.

Andando ad osservare quanto è contenuto in A', possiamo verificare che la (1), la (2) e la (3) sono vere. Infatti, non possiamo concludere Casaro(Davide) visto che non sappiamo se Asiago sia un Formaggio. Avendo introdotto due anonimi - ovvero X1 e X2 - la (5) è vera, mentre la (4) e la (6) sono false.

9. Esercizio Truth tables e entailment in LOP (5 punti)

Indicare quali delle seguenti affermazioni riguardanti le seguenti formule in linguaggio LOP sono vere (una o più).

P: $(X \wedge Y) \vee (\neg X \wedge Y) \vee (\neg X \wedge \neg Z)$
A: $Y \vee \neg Z$
B: $X \wedge \neg Z$
C: $(\neg X \wedge \neg Y \wedge \neg Z) \vee Y \vee Z$

1. A è conseguenza logica di P
2. B è conseguenza logica di P
3. C è conseguenza logica di P
4. A è equivalente a P
5. B è equivalente a P
6. C è equivalente a P

SOLUZIONE: La tabella di verità di P e di A, B e C è la seguente:

X	Y	Z	P	A	B	C
F	F	F	**T**	F	F	**T**
F	F	T	F	F	F	**T**
F	T	F	**T**	**T**	F	**T**
F	T	T	**T**	F	F	**T**
T	F	F	F	F	**T**	F
T	F	T	F	F	F	**T**
T	T	F	**T**	**T**	**T**	**T**
T	T	T	**T**	F	F	**T**

Si ricorda che una formula Q è conseguenza logica di P se tutti i modelli di P sono anche modelli di Q. P e Q sono equivalenti se hanno esattamente gli stessi modelli.

Dalla tabella si evince chiaramente che A e C sono conseguenze logiche di P, mentre B non lo è. Nessuna delle A, B, C è equivalente a P. Di conseguenza, sono vere solo (1) e (3).

10. Esercizio CNF (2 punti)

Convertire in CNF la seguente formula in logica delle proposizioni (LOP).

$A \wedge (A \equiv (B \supset \neg D)) \wedge \neg(C \supset (C \wedge A))$

Dire quindi, quale di queste è il risultato dell'algoritmo di conversione.

1. $A \wedge (A \vee B) \wedge (\neg A \vee \neg B \vee \neg D) \wedge (B \vee D) \wedge (C \vee \neg A)$
2. $A \wedge (A \vee \neg B \vee \neg D) \wedge (B \vee D) \wedge (B \vee A) \wedge (C \vee \neg A)$
3. $A \wedge C \wedge (A \vee B) \wedge (\neg A \vee \neg B \vee \neg D) \wedge (A \vee D) \wedge (C \vee \neg A)$
4. $A \wedge C \wedge (\neg A \vee B) \wedge (\neg A \vee \neg B \vee \neg D) \wedge (B \vee D) \wedge (\neg C \vee \neg A)$

SOLUZIONE.

La risposta corretta è la 3. Infatti:

$CNF(A \wedge (A \equiv (B \supset \neg D)) \wedge \neg(C \supset (\neg C \wedge A)) =$

$\mathbf{CNF(A)} \wedge \mathbf{CNF(A \equiv (B \supset \neg D))} \wedge \mathbf{CNF(\neg(C \supset (\neg C \wedge A)))}$

Calcoliamo separatamente le 3 CNF:

CNF(A) = A.

$\mathbf{CNF(A \equiv (B \supset \neg D))} = CNF(A \supset (B \supset \neg D)) \wedge CNF((B \supset \neg D) \supset A) =$

$(CNF(\neg A) \otimes CNF(B \supset \neg D)) \wedge CNF((B \supset \neg D) \supset A) =$

$(CNF(\neg A) \otimes CNF(\neg B) \otimes CNF(\neg D)) \wedge CNF((B \supset \neg D) \supset A) =$

$(\neg A \vee \neg B \vee \neg D) \wedge CNF((B \supset \neg D) \supset A) =$

$(\neg A \vee \neg B \vee \neg D) \wedge CNF(\neg(B \supset \neg D) \otimes CNF(A)) =$

$(\neg A \vee \neg B \vee \neg D) \wedge ((CNF(B) \wedge CNF(\neg\neg D)) \otimes CNF(A)) =$

$(\neg A \vee \neg B \vee \neg D) \wedge ((B \wedge D) \otimes A) =$

$(\neg A \vee \neg B \vee \neg D) \wedge (B \vee A) \wedge (D \vee A)$.

$CNF(\neg(C \supset (\neg C \wedge A)))$ =

$CNF(C) \wedge CNF(\neg(\neg C \wedge A))$ =

$C \wedge (CNF(C) \otimes CNF(\neg A))$ =

$C \wedge (C \vee \neg A)$

Di conseguenza la soluzione è $A \wedge (\neg A \vee \neg B \vee \neg D) \wedge (B \vee A) \wedge (D \vee A) \wedge C \wedge (C \vee \neg A)$

11. Esercizio DPLL (4 punti)

Data la seguente formula ϕ in logica delle proposizioni (LOP), indicare quali delle seguenti affermazioni sono vere (una o più).

{{P, Q}, {¬Q, P}, {¬P, Q}, {¬P, ¬Q, R}, {¬P, R}}

1. La formula ϕ è soddisfacibile
2. La formula ϕ è insoddisfacibile
3. R, P, Q è una possibile sequenza di assegnamenti generati dall'applicazione del DPLL
4. R, P, ¬Q è una possibile sequenza di assegnamenti generati dall'applicazione del DPLL
5. P, R, Q è una possibile sequenza di assegnamenti generati dall'applicazione del DPLL

SOLUZIONE

Applichiamo quindi il DPLL a: {{P, Q}, {¬Q, P}, {¬P, Q}, {¬P, ¬Q, R}, {¬P, R}}

Osserviamo che non ci sono unit clause. R però appare come pure literal.

{{P, Q}, {¬Q, P}, {¬P, Q}, {¬P, ¬Q, ⊤}, {¬P, ⊤}}

{{P, Q}, {¬Q, P}, {¬P, Q}}

Osserviamo che non ci sono unit clause né pure literals. Scegliamo quindi ad esempio P per applicare il branching.

{{⊤, Q}, {¬Q, ⊤}, {⊥, Q}}

{{Q}}

Adesso abbiamo la unit clause Q.

{{⊤}} che genera {}.

Di conseguenza, restituisce true. Pertanto, sono vere la (1) e la (3).

Si può anche verificare in https://www.inf.ufpr.br/dpasqualin/d3-dpll/ con la sequenza:

1 2

-2 1

-1 2

-1 -2 3

-1 3

ESERCIZI AVANZATI DA COMPITI PRECEDENTI E NON ANCORA USATI

1.

8. Esercizio Basic facts reasoning (3 punti)

Utilizzando le proprietà della logica delle proposizioni in linguaggio LOP indicare quale delle seguenti affermazioni è vera.

1. **$((p \supset q) \supset q) \supset q \equiv p \supset q$ è vera**
2. **$((p \supset q) \supset q) \supset q \equiv p \supset q$ è falsa**

NOTA: si suggerisce di riscrivere la formula di sinistra, usando le proprietà della logica, sino a quando non si giunge alla formula di destra, oppure sino a quando le formule ottenute sono chiaramente diverse.

SOLUZIONE:

La soluzione è la (1) perché:

$((p \supset q) \supset q) \supset q$	$\equiv ((\neg p \vee q) \supset q) \supset q$	Implication and disjunction
	$\equiv (\neg(\neg p \vee q) \vee q) \supset q$	Implication and disjunction
	$\equiv ((p \wedge \neg q) \vee q) \supset q$	De Morgan

$\equiv ((p \vee q) \wedge (\neg q \vee q)) \supset q$ Associative

$\equiv (p \vee q) \supset q$ The law of excluded middle

$\equiv (p \supset q) \wedge (q \supset q)$ Implication and disjunction

$\equiv (p \supset q) \wedge (\neg q \vee q)$ Implication and disjunction

$\equiv p \supset q$ The law of excluded middle

10. Esercizio entailment in LODE (5 punti)

Data la seguente TBOX T in linguaggio LODE:

Parent ≡ Person ⊓ ∃hasChild.Person

HappyParent ≡ Parent ⊓ ∃hasChild.Doctor

LuckyParent ≡ Parent ⊓ ∀hasChild.Doctor

e la seguente ABOX A in linguaggio LODE:

Person(Lucas)

LuckyParent(Mary)

HappyParent(Bob)

hasChild(Mary, Bob)

hasChild(Bob, Lucas)

Denotiamo inoltre con EG il LODE Entity Graph che risulta dalla composizione, attraverso lo sviluppo ("unfolding") della TBOX e espansione ("expansion") della ABOX.

Indicare quale delle seguenti affermazioni sono vere (una o più):

1. **T ⊨ Person(Bob)**
2. **T ⊨ Doctor(Bob)**
3. **T ⊨ Doctoy(Lucas)**

4. **T ⊨ hasChild(Mary, Lucas)**
5. **L’EG contiene un solo nodo che rappresenta entità anonime**
6. **L’EG contiene esattamente due nodi che rappresentano entità anonime**
7. **L’EG contiene più di due nodi che rappresentano entità anonime**

SOLUZIONE.

Cominciamo a calcolare l’unfolding della TBOX:

Parent ≡ Person ⊓ ∃hasChild.Person

HappyParent ≡ Person ⊓ ∃hasChild.Person ⊓ ∃hasChild.Doctor

LuckyParent ≡ Person ⊓ ∃hasChild.Person ⊓ ∀hasChild.Doctor

Calcoliamo quindi l’espansione A’ della ABOX rispetto alla TBOX unfolded:

Person(Lucas)

LuckyParent(Mary), hasChild(Mary, Bob), Person(Mary), hasChild(Mary, X1), Person(X1), Doctor(X1), Doctor(Bob)

HappyParent(Bob), Person(Bob), hasChild(Bob, X2), Person(X2), hasChild(Bob, X3), Doctor(X3)

hasChild(Bob, Lucas)

Andando ad osservare quanto è contenuto in A’, possiamo verificare che la (1) e la (2) sono vere, mentre la (3) e (4) sono false. Avendo introdotto tre anonimi - ovvero X1, X2 e X3 - la (7) è vera, mentre la (5) e la (6) sono false.