Logica Computazionale

Final Term

20 Dicembre 2024

TOTALE: 36pt

Durata 120 minuti

Lista esercizi (max 36pt)

1. **Reasoning in LoDE (4pt).**
2. **Representations – world models, world logics, language logics (3pt).**
3. **Proprietà del logical entailment (3pt).**
4. **LoP – domain, model, facts, percepts, interpretation (5pt).**
5. **LoP + LoI – Reasoning problems (2pt).**
6. **Lop + LoI – Informal to Formal (1pt).**
7. **Lop + LoI – Informal to Formal (1pt).**
8. **Lop + LoI – Informal to Formal (2pt).**
9. **LoP + LoI – proprietà dei connettivi logici (5pt).**
10. **Reasoning (3pt).**
11. **DPLL – reasoning (4pt).**
12. **LoP to CNF (3pt).**

1. Reasoning in LoDE (4pt)

Considera le seguenti TBox e ABox. Indica quali delle seguenti affermazioni sono vere (una o più).

RestaurantEmployee ≡ Person ⊓ ∀worksAt.Restaurant

RestaurantOwner ≡ Person ⊓ ∀ownsA.Restaurant

Cook ≡ RestaurantEmployee ⊓ ∃canPrepare.Recipe

Waiter ≡ RestaurantEmployee ⊓ ∃serve.Table

Chef ≡ Cook ⊓ ∃headOf.Kitchen

Waiter⊥Cook

RestaurantEmployee⊥RestaurantOwner

Chef(Mike#1)
Waiter(Will#1)
RestaurantOwner(Robert#1)

1. Espandendo Chef(Mike#1) si aggiungono due nuove entità alla ABox.
2. Espandendo RestaurantOwner(Robert#1) si aggiunge una nuova entità alla ABox.
3. Waiter ⊑ ¬RestaurantOwner
4. Cook ⊑ ¬∃ownsA.¬Restaurant
5. Chef ⊓ Waiter ≡ ⊥

SOLUZIONE

RestaurantEmployee ≡ Person ⊓ ∀worksAt.Restaurant
RestaurantOwner ≡ Person ⊓ ∀ownsA.Restaurant
Cook ≡ Person ⊓ ∀worksAt.Restaurant ⊓ ∃canPrepare.Recipe
Waiter ≡ Person ⊓ ∀worksAt.Restaurant ⊓ ∃serve.Table
Chef ≡ Person ⊓ ∀worksAt.Restaurant ⊓ ∃canPrepare.Recipe ⊓ ∃headOf.Kitchen

exp(Chef(Mike#1))

- **exp(Chef(Mike#1)) = {Chef(Mike#1), exp(Cook(Mike#1)), exp(∃headOf.Kitchen(Mike#1))}**

- exp(Chef(Mike#1)) = {Chef(Mike#1), Cook(Mike#1), exp(RestaurantEmployee(Mike#1)), exp(∃canPrepare.Recipe(Mike#1)), headOf(Mike#1, Anonym#1), Kitchen(Anonym#1)}
- exp(Chef(Mike#1)) = {Chef(Mike#1), Cook(Mike#1), RestaurantEmployee(Mike#1), exp(Person(Mike#1)), exp(∀worksAt.Restaurant(Mike#1)), canPrepare(Mike#1,Anonym#2), Recipe(Anonym#2), headOf(Mike#1, Anonym#1), Kitchen(Anonym#1)}
- exp(Chef(Mike#1)) = {Chef(Mike#1), Cook(Mike#1), RestaurantEmployee(Mike#1), Person(Mike#1), canPrepare(Mike#1,Anonym#2), Recipe(Anonym#2), headOf(Mike#1, Anonym#1), Kitchen(Anonym#1)}

exp(RestaurantOwner(Robert#1))

- exp(RestaurantOwner(Robert#1)) = {RestaurantOwner(Robert#1), exp(Person(Robert#1)), exp(∀ownsA.Restaurant(Robert#1))}
- exp(RestaurantOwner(Robert#1)) = {RestaurantOwner(Robert#1), Person(Robert#1)}

1. Vero, l'espansione del quantificatore esistenziale richiede in questo caso l'introduzione di due nuove entità anonime.
2. Falso, l'espansione del quantificatore esistenziale non richiede mai l'introduzione di nuove entità anonime
3. Vero
4. Falso
5. Vero

2. Representations – world models, world logics, language logics (3pt).

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

1. I modelli del mondo ("world models") ci permettono di stabilire se una asserzione è vera nel modello inteso.
2. In LoDE, è possibile distinguere fra quando un fatto non si sa se è vero e quando si sa che è vero
3. In un modello di LoP, è possibile distinguere fra quando un fatto, come rappresentato da una proposizione, è vero e quando è falso in quel modello
4. In LoDE, è possibile distinguere fra quando non si sa se un fatto è vero o falso e quando si sa che è falso
5. In LoP è possibile distinguere fra quando non si sa se un fatto è vero o falso e quando si sa che è falso

SOLUZIONE

1. **Falso, i modelli del mondo (ad esempio un ER model) non permettono di asserire la verità di una asserzione o teoria. Essi possono essere usati solo per affermare (usando una semantica formale od informale) se rappresentazione linguistica ed una analogica sono la prima una descrizione corretta della seconda (slides "2024-10-08 HP2T - World MRLA", "2024-11-21 HP2T – Logics"**
2. **Vero, una asserzione asserisce la verità del fatto che denota.**
3. **Vero, la verità / falsità di una proposizione asserisce la verità o la falsità del fatto LoDE di cui esprime il "judgement"**
4. **Falso, in LoDE non si sa niente dei fatti per cui esiste una asserzione che ne asserisce la verità**
5. **Vero, La proposizione associata ad una fatto vero (falso) è vera (falsa) in tutti i modelli della teoria. Per gli altri esistono modelli in cui sono vere ed altri in cui sono false.**

3. Proprietà del logical entailment (3pt)

Logical entailment è la formalizzazione del ragionamento logico nella logica LoP. Scriverne le proprietà, nome e formula, una per riga. (NOTA: il numero di righe non corrisponde necessariamente al numero di proprietà)

SOLUZIONE

1. **Reflexivity: w |= w**
2. **Cut: If T |= w_1 and Σ ∪ {w_1} |= w_2 then T ∪ Σ |= w_2**
3. **Compactness: If T |= w then there is a finite subset $T_0 \subseteq T$ such that T_0 |= w**
4. **Monotonicity: If T |= w then T ∪ Σ |= w**
5. **NonMonotonicity: If T |= w and T ∪ Σ then not|= w**

4. LoP – domain, model, facts, percepts, interpretation (5pt).

Si consideri la logica LoP. Indicare quali delle seguenti affermazioni sulle logiche sono VERE (una o più):

1. Un modello LoP contiene esattamente la metà dei percetti di un dominio LoP
2. In LoP la falsità di una asserzione A appartenente al modello inteso LoDE può essere asserita solo usando la negazione, ossia una formula del tipo “not P” dove P è una proposizione che asserisce la verità di A
3. Se A è una asserzione vera in un modello LoDE, allora la proposizione che asserisce la verità di A (“It is true that A”), se parte del linguaggio di LoP, deve essere vera nel modello LoP
4. Una teoria LoP ha sempre uno o più modelli
5. Un modello LoP è completamente determinato dalle proposizioni atomiche false che contiene
6. Tutte le teorie LoP hanno un modello minimo
7. Se una teoria T2 estende una teoria T1 con una formula in più(un assioma) allora T2 ha tanti modelli quanto T1 o di più
8. Il valore di verità di una formula A dipende solo dal valore di verità delle proposizioni atomiche che occorrono in A.

SOLUZIONE

1. **Vero**
2. **Falso, può essere asserita con la proposizione P che dice “It is false that A”**
3. **Vera**
4. **Falso, una teoria inconsistente non ha modelli**
5. **Vero, la conoscenza della formule atomiche false permette di stabilire (per differenza) quelle vere.**
6. **Falso, la teoria che contiene l’assioma “A or B” non ha modello minimo.**
7. **Falso, tanto più assiomi si aggiungono tanto più si mettono vincoli ai modelli della teoria. Basti pensare alle teorie T1= {A}, T2={A,B}. I Modelli dove B è falso sono modelli di T1 ma non di T2.**
8. **Vero**

5. LoP + LoI – Reasoning problems (2pt).

Si considerino le logiche LoP e LoI. Indicare quali delle seguenti affermazioni sulle logiche sono vere (una o più):

1. Se una formula è soddisfacibile allora la sua negata è insoddisfacibile
2. Una formula è soddisfacibile od insoddisfacibile
3. In LoP il problema della equivalenza logica può essere ridotto ad un problema di insoddisfacibilità
4. In LoI, una formula con una variabile libera è soddisfacibile se e solo se tutte le assegnazioni ("assignment") di costanti alla variabile libera generano formule che sono soddisfacibili.

SOLUZIONE

1. **Falso, la sua negata puo' essa stessa soddisfacibile Ad esempio le formule "A" e "not A" sono entrambe soddisfacibili**
2. **Vero**
3. **Vero**
4. **Falso, basta che esista una assegnazione (si veda la lezione "2024-12-06.05 HP2T – LOI", slide 73)**

Date il seguente mapping di testo informale in simboli:

- “A Luigi piace sciare” -> LS
- “Luigi va in montagna” -> LM
- “Luigi va in piscina” -> LP
- “A Nicole piace sciare” -> NS
- “Nicole va in montagna” -> NM
- “Nicole va in piscina” -> NP
- “Piace” -> L
- “Sciare” -> S
- “Va in” -> G
- “Montagna” -> M

6. LoP + LoI - informal to formal (1pt)

Indicare quale singolo connettivo logico deve essere utilizzato nella traduzione in logica delle proposizioni (LOP) della frase

“Nicola va in montagna o in piscina”.

1. ∧ (and)
2. ∨ (or)
3. ¬ (not)
4. + (xor)
5. ⊃ (implicazione)
6. ≡ (equivalenza)
7. Nessun connettivo logico

SOLUZIONE

- **Xor**

7. LoP + LoI - informal to formal (1pt)

Indicare quale singolo connettivo logico deve essere utilizzato nella traduzione in logica delle proposizioni (LOP) della frase

“Luigi e Nicole sono in montagna assieme”.

1. ∧ (and)
2. ∨ (or)

3. ¬ (not)
4. + (xor)
5. ⊃ (implicazione)
6. ≡ (equivalenza)
7. Nessun connettivo logico

SOLUZIONE

- **Nessun connettivo logico**

8. LoP + LoI - informal to formal (2pt)

Formalizza la seguente frase in LoI: “Esiste solo una persona che va in montagna e a cui piace sciare”.

SOLUZIONE

La formalizzazine deve esprimenre la congiunzione di due fatti veri, il primo che dice che esiste una persona, il secondo che dice che è una sola. Ci sono varie formalizzazioni. Sotto ne abbiamo indicate alcune, ma varie altre esistono

$$\exists x.\,(G(x,M) \wedge L(x,S)) \wedge \forall y.\forall z.\,((G(y,M) \wedge L(y,S) \wedge G(z,M) \wedge L(z,S)) \supset (y = z)))$$

$$\exists x.\,(G(x,M) \wedge L(x,S) \wedge \forall y.\,((G(y,M) \wedge L(y,S)) \supset (y = x)))$$

9. LoP + LoI proprietà dei connettivi logici (5pt)

Indica quali delle seguenti affermazioni sono vere.

1. La formula seguente è valida in LoP
   $((A \vee \mathrm{B}) \wedge \mathrm{C}) \supset \mathrm{D} \equiv (A \vee \mathrm{B}) \supset (\mathrm{D} \vee \neg \mathrm{C})$

2. La formula seguente è valida in LoP
   $(A \wedge \mathrm{C}) \supset \mathrm{D} \equiv A \vee \mathrm{B} \supset (\mathrm{D} \vee \neg \mathrm{C})$

3. La formula seguente è soddisfacibile in LoI
   $\forall \mathrm{x} .(\mathrm{A}(\mathrm{x}) \vee \mathrm{B}(\mathrm{x})) \equiv \forall \mathrm{x}.\mathrm{A}(\mathrm{x}) \vee \forall \mathrm{x}.\mathrm{B}(\mathrm{x})$

4. La formula seguente è valida in LoI
   $\exists \mathrm{x}. (\mathrm{A}(\mathrm{x}) \wedge \mathrm{B}(\mathrm{x})) \equiv \exists \mathrm{x}. \mathrm{A}(\mathrm{x}) \wedge \exists \mathrm{x}. \mathrm{B}(\mathrm{x})$

5. La formula seguente è valida in LoI
   $\forall \mathrm{x} . \exists \mathrm{y}. \mathrm{A}(\mathrm{x}, \mathrm{y}) \equiv \exists \mathrm{y}. \forall \mathrm{x}. \mathrm{A}(\mathrm{x}, \mathrm{y})$

SOLUZIONE

1. **Vero - è una delle proprietà fondamentali dell'interazione fra implicazione, congiunzione e disgiunzione. Per vederlo basta sostituire "$(A \wedge B)$" con "A". (si veda la lezione "2024-11-28.22 HP2T – LOP")**
2. **Falso, e' solo soddisfacibile. Per vederlo basta eliminare " $\vee$ B". La formula risultante è una tautologia. (si veda la lezione "2024-11-28.22 HP2T – LOP")**
3. **Vera, basta scegliere un modello dove A è vera per tutte le x. (si veda la lezione "2024-12-06.05 HP2T – LOI")**
4. **Falsa, vera in una sola direzione, è solo soddisfacibile. (si veda la lezione "2024-12-06.05 HP2T – LOI")**
5. **Falsa, vera in una sola direzione è solo soddisfacibile. (si veda la lezione "2024-12-06.05 HP2T – LOI")**

10. Reasoning (3pt)

Indica quali delle seguenti affermazioni sono vere.

1. Una formula in CNF è anche in NNF
2. Ci sono formule che sono sia in forma CNF che in forma DNF
3. in LoI, dato un dominio finito non ci può essere un insieme infinito di costanti che descrive tutte gli elementi del dominio
4. Dato un dominio infinito, un linguaggio LoI non può avere nomi per tutti gli elementi del dominio
5. Una formula LoI quantificata universalmente può essere ridotta ad una formula proposizionale ma la formula proposizionale risultante è espansa di una fattore $n^{m,}$ dove n è la dimensione del dominio e m è il numero di variabili quantificate.

SOLUZIONE

1. **Vero, CNF è un caso particolare di NNF**
2. **Vero, una congiunzione di literals come anche una disgiunzione di literals sono sia in forma CNF che DNF**
3. **Falso, basta che i nomi siano tutti sinonimi**
4. **Falso, basta che ci sia un simbolo funzionale (si veda la lezione "2024-12-06.05 HP2T – LOI")**
5. **Falso, n è il numero di variabili quantificate e m è la dimensione del dominio**

11. DPLL (4pt)

Si consideri la procedura DPLL (versione finale) come da lezione. Data la seguente formula in CNF, indicare quale delle sequenti affermazioni sono vere (una o più).

$$(\boldsymbol{a} \vee \neg \boldsymbol{b} \vee \neg \boldsymbol{c}) \wedge (\neg \boldsymbol{a} \vee \neg \boldsymbol{b} \vee \boldsymbol{d}) \wedge (\neg \boldsymbol{a} \vee \boldsymbol{b} \vee \neg \boldsymbol{d}) \wedge \neg \boldsymbol{c}$$

1. $\{b, d, \neg \boldsymbol{c}\}$ è uno dei modelli possibili ritornati da DPLL
2. $\{b, d, \neg \boldsymbol{c}\}$ è un modello della formula non ritornato da DPLL
3. $\{\neg \boldsymbol{c}, \neg \boldsymbol{a}\}$ è un modello ritornato da DPLL
4. $\{\neg \boldsymbol{c}, \neg \boldsymbol{a}\}$ è l'unico modello ritornato da DPLL
5. DPLL non esegue alcun pure-literal assignment.

SOLUZIONE

$$(\boldsymbol{a} \vee \neg \boldsymbol{b} \vee \neg \boldsymbol{c}) \wedge (\neg \boldsymbol{a} \vee \neg \boldsymbol{b} \vee \boldsymbol{d}) \wedge (\neg \boldsymbol{a} \vee \boldsymbol{b} \vee \neg \boldsymbol{d}) \wedge \neg \boldsymbol{c}$$

$$\big\{\{\boldsymbol{a}, \neg \boldsymbol{b}, \neg \boldsymbol{c}\}, \{\neg \boldsymbol{a}, \neg \boldsymbol{b}, \boldsymbol{d}\}, \{\neg \boldsymbol{a}, \boldsymbol{b}, \neg \boldsymbol{d}\}, \neg \boldsymbol{c}\big\} \;\; \boldsymbol{I}: \{\}$$

$$\big\{\{\boldsymbol{a}, \neg \boldsymbol{b}, \neg \boldsymbol{c}\}, \{\neg \boldsymbol{a}, \neg \boldsymbol{b}, \boldsymbol{d}\}, \{\neg \boldsymbol{a}, \boldsymbol{b}, \neg \boldsymbol{d}\}, \neg \boldsymbol{c}\big\}|_{\neg c} \; (\boldsymbol{unit\ propagation}) \; \boldsymbol{I}: \{\neg \boldsymbol{c}\}$$

$$\big\{\{\boldsymbol{a}, \neg \boldsymbol{b}, \boldsymbol{V}\}, \{\neg \boldsymbol{a}, \neg \boldsymbol{b}, \boldsymbol{d}\}, \{\neg \boldsymbol{a}, \boldsymbol{b}, \neg \boldsymbol{d}\}, \boldsymbol{V}\big\} \; \boldsymbol{I}: \{\neg \boldsymbol{c}\}$$

$$\big\{\{\neg \boldsymbol{a}, \neg \boldsymbol{b}, \boldsymbol{d}\}, \{\neg \boldsymbol{a}, \boldsymbol{b}, \neg \boldsymbol{d}\}\big\}|_{\neg a} (\boldsymbol{pure\ literal}) \;\; \boldsymbol{I}: \{\neg \boldsymbol{c}, \neg \boldsymbol{a}\}$$

$$\big\{\{\boldsymbol{V}, \neg \boldsymbol{b}, \boldsymbol{d}\}, \{\boldsymbol{V}, \boldsymbol{b}, \neg \boldsymbol{d}\}\big\} \;\; \boldsymbol{I}: \{\neg \boldsymbol{c}, \neg \boldsymbol{a}\}$$

$$\{\} \; \boldsymbol{I}: \{\neg \boldsymbol{c}, \neg \boldsymbol{a}\}$$

1. **F – considerando la versione finale della procedura DPLL, vengono eseguite prima le sostituzioni dovute da unit propagation e pure literal, e dopodichè le sostituzioni dovute ad una euristica, nel caso di questa formula, DPLL la verifica con solo due sostituzioni obbligate (come mostrato sopra), di conseguenza $\{b, d, \neg c\}$ non verrà mai ritornato da DPLL, anche se ne è un modello.**
2. **V – vedi sopra.**
3. **V – vedi procedimento sopra.**
4. **V – vedi sopra e anche procedimento.**
5. **F – dal procedimento sopra si può vedere come viene eseguito un pure literal assignment.**

12. LoP to CNF (3pt)

Converti la seguente formula in LoP in CNF.

$$\neg((\boldsymbol{a} \supset (\boldsymbol{b} \vee \boldsymbol{c})) \supset ((\boldsymbol{b} \wedge \boldsymbol{a}) \supset \neg(\boldsymbol{c} \vee \boldsymbol{b})))$$

La formula finale va scritta seguendo l'ordine lessico-grafico sia per quanto riguarda l'ordine dei literals all'interno delle clausole e quello delle clausole fra di loro (note: nell'ordinamento i letterali negati vengono dopo quelli non negati).

The final formula must be written following the lexico-graphic ordering in the writing of both clauses and literals inside clauses (note: in the ordering negated literals come after their positive counterpart).

SOLUZIONE

$$\boldsymbol{CNF}(\neg((\boldsymbol{a} \supset (\boldsymbol{b} \vee \boldsymbol{c})) \supset ((\boldsymbol{b} \wedge \boldsymbol{a}) \supset \neg(\boldsymbol{c} \vee \boldsymbol{b}))))$$
$$\boldsymbol{CNF}(\boldsymbol{a} \supset (\boldsymbol{b} \vee \boldsymbol{c})) \wedge \boldsymbol{CNF}(\neg((\boldsymbol{b} \wedge \boldsymbol{a}) \supset \neg(\boldsymbol{c} \vee \boldsymbol{b})))$$
$$(\boldsymbol{CNF}(\neg\boldsymbol{a}) \otimes \boldsymbol{CNF}(\boldsymbol{b} \vee \boldsymbol{c})) \wedge (\boldsymbol{CNF}(\boldsymbol{b} \wedge \boldsymbol{a}) \wedge \boldsymbol{CNF}(\boldsymbol{c} \vee \boldsymbol{b}))$$
$$(\boldsymbol{CNF}(\neg\boldsymbol{a}) \otimes (\boldsymbol{CNF}(\boldsymbol{b}) \otimes \boldsymbol{CNF}(\boldsymbol{c}))) \wedge (((\boldsymbol{CNF}(\boldsymbol{b}) \wedge \boldsymbol{CNF}(\boldsymbol{a})) \wedge (\boldsymbol{CNF}(\boldsymbol{c}) \otimes \boldsymbol{CNF}(\boldsymbol{b})))$$
$$(\neg\boldsymbol{a} \otimes (\boldsymbol{b} \otimes \boldsymbol{c})) \wedge ((\boldsymbol{b} \wedge \boldsymbol{a}) \wedge (\boldsymbol{c} \otimes \boldsymbol{b}))$$
$$(\neg\boldsymbol{a} \vee \boldsymbol{b} \vee \boldsymbol{c}) \wedge \boldsymbol{b} \wedge \boldsymbol{a} \wedge (\boldsymbol{c} \vee \boldsymbol{b})$$

$$\boldsymbol{a} \wedge \boldsymbol{b} \wedge (\boldsymbol{b} \vee \boldsymbol{c}) \wedge (\neg\boldsymbol{a} \vee \boldsymbol{b} \vee \boldsymbol{c})$$