Logica per l'Informatica – Luglio 2019

1. Sia F la formula $\forall x(p(x) \to \exists y(q(y) \wedge r(x,y)))$. Definire un'interpretazione $\mathcal{M}$ con $\mathcal{M}(q) = \emptyset$ e tale che $\mathcal{M}$ sia un modello di F. Dimostrare formalmente che $\mathcal{M} \models F$.

2. Dimostrare mediante risoluzione che:

$$\forall x\,(\neg\forall y\, p(x,y) \to q(x)),\ \neg\exists x\,(r(x) \wedge q(x)),$$
$$\forall x\,(\forall y\, p(x,y) \to \neg s(x)) \quad \vdash_{Res} \forall x\,(r(x) \to \neg s(x))$$

   (Attenzione a leggere le parentesi). Dopo aver costruito l'insieme di clausole per la dimostrazione (<u>mostrandone tutti i passaggi</u>), spiegare perché la risoluzione SLD non si può applicare all'insieme di clausole ottenute. Applicare invece la strategia di risoluzione lineare per derivare la clausola vuota (indicando ad ogni passaggio la sostituzione applicata).

3. Definire un predicato Prolog `disordinati(+L,?D)` che ha successo se e solo se `L` è una lista di interi e `D` è una lista che contiene gli elementi di `L` che sono minori dell'elemento precedente. Gli elementi di `D` devono occorrere nello stesso ordine in cui occorrono in `L`. Ovviamente, se `L` è ordinata, `D` sarà vuota.

   Ad esempio, `disordinati([1,10,5,6,7,4,9],Result)` avrà successo con `Result = [5,4]`. E `disordinati([1,2,3,4,5,6],Result)` avrà successo con `Result = []` (se la lista `L` è ordinata, `D` è vuota).

4. Sia $G = \Diamond(p \wedge \Box(q \vee \neg p))$. Costruire un tableau completo per la formula G. Identificare l'insieme f_G dei nodi di accettazione di G e caratterizzare i cammini aperti del tableau. Rappresentare graficamente un automa di Büchi $\mathcal{A}$ che accetti tutti e solo i modelli di G. Nel tableau numerare i nodi e utilizzare la stessa numerazione nella rappresentazione dell'automa.

   Determinare, infine, un'esecuzione di accettazione ρ dell'automa $\mathcal{A}$ e una parola $\mathcal{M}$ letta da ρ (quindi accettata da $\mathcal{A}$).