8 maggio 2020 - lezione 1

Avvertenza importante!

Questo pdf contiene indicativamente il materiale della prima ora di lezione di venerdì 8 maggio 2020. La forma è volutamente discorsiva, cercando di imitare lo stile che avrei utilizzato nella lezione “dal vivo”. Per lo studio, si raccomanda di utilizzare gli appunti di “Logica” disponibili in rete nella pagina e-learning (“Moodle”) di questo insegnamento. Il materiale contenuto negli appunti è più che sufficiente per lo studio della materia!

In questa prima ora (virtuale) di lezione di oggi venerdì 8 maggio 2020 vediamo lo svolgimento dei due esercizi che vi avevo proposto lo scorso martedì 5 maggio 2020, rispettivamente nella prima e nella seconda ora (virtuale) di lezione.

Parlando dell’uso di formule aperte di un linguaggio di logica dei predicati per formalizzare efficacemente le definizioni, nella prima ora vi avevo chiesto di scrivere

– una formula aperta $\varphi(x)$ che risulti vera sse x è un divisore dello zero;

– una formula aperta $\psi(x)$ che risulti vera sse x è invertibile.

scegliendo un opportuno linguaggio della logica dei predicati e un’opportuna interpretazione in un anello commutativo con unità.

Per esprimere il fatto che x è un divisore dello zero ci servono:

(i) un simbolo di funzione binaria, p, che interpreteremo nella funzione *prodotto* dell’anello (la funzione che a ogni coppia ordinata di elementi dell’anello associa il loro prodotto);

(ii) un simbolo di predicato binario, U, che interpreteremo nel predicato “è uguale a”;

(iii) un simbolo di costante, c, che interpreteremo nello *zero* dell’anello (che è l’elemento neutro per la somma).

In questo linguaggio di logica dei predicati, con l’interpretazione sopra esposta in qualsiasi anello commutativo (e quindi in particolare anche nell’anello $\mathbb{Z}_n$ per qualsiasi valore di $n \in \mathbb{N}\setminus\{0,1\}$) la formula aperta

$$\varphi(x) := \neg \mathrm{U}(x,c) \wedge (\exists y)\big(\neg \mathrm{U}(y,c) \wedge \mathrm{U}\big(p(x,y),c\big)\big)$$

risulta vera esattamente per quelle assegnazioni di valore che assegnano a x un divisore dello zero.

Nello stesso linguaggio di logica dei predicati possiamo anche esprimere il fatto che x è invertibile (in un anello commutativo con unità), interpretando però questa volta la costante c nell’unità dell’anello (che è l’elemento neutro del prodotto). La formula aperta

$$\psi(x) := (\exists y)\mathrm{U}\big(p(x,y),c\big)$$

risulta vera esattamente per quelle assegnazioni di valore che assegnano a x un elemento invertibile dell’anello.

L’esercizio proposto nella seconda ora era invece il seguente: si stabilisca se dalle premesse

(i) Ogni cane è un mammifero;
(ii) il padre di ogni cane è un cane;
(iii) Alf non è un mammifero;

si può dedurre logicamente che

(iv) Alf non è il padre di alcun cane.

Per valutare la correttezza di questo ragionamento, consideriamo un linguaggio di logica dei predicati con due simboli di predicato unario (C, M), un simbolo di predicato binario (P) e un simbolo di costante (a), che interpretiamo come segue per tradurre le premesse date:

C(x) := x è un cane;
M(x) := x è un mammifero;
P(x, y) := x è il padre di y;
a := Alf.

In questo linguaggio della logica dei predicati, le premesse diventano:

(i) $(\forall x)\big(\mathrm{C}(x) \to \mathrm{M}(x)\big)$;
(ii) $(\forall x)(\forall y)\big(\big(\mathrm{C}(x) \wedge \mathrm{P}(y,x)\big) \to \mathrm{C}(y)\big)$;
(iii) $\neg \mathrm{M}(a)$;

e la conclusione diventa

(iv) $\neg(\exists x)\big(\mathrm{C}(x) \wedge \mathrm{P}(a,x)\big)$.

Dobbiamo pertanto vedere se

$$(\forall x)(\mathrm{C}(x) \to \mathrm{M}(x)) \wedge (\forall x)(\forall y)((\mathrm{C}(x) \wedge \mathrm{P}(y,x)) \to \mathrm{C}(y)) \wedge \neg\mathrm{M}(a) \vDash \neg(\exists x)(\mathrm{C}(x) \wedge \mathrm{P}(a,x))$$

e questo è vero se e soltanto se la formula φ definita da

$$\varphi \coloneqq (\forall x)(\mathrm{C}(x) \to \mathrm{M}(x)) \wedge (\forall x)(\forall y)((\mathrm{C}(x) \wedge \mathrm{P}(y,x)) \to \mathrm{C}(y)) \wedge \neg\mathrm{M}(a) \wedge (\exists x)(\mathrm{C}(x) \wedge \mathrm{P}(a,x))$$

non è soddisfacibile.

Scriviamo φ in forma normale prenessa in modo che la parte della formula che segue i quantificatori sia in forma normale congiuntiva:

$$\varphi \coloneqq (\forall x)(\neg\mathrm{C}(x) \vee \mathrm{M}(x)) \wedge (\forall x)(\forall y)(\neg(\mathrm{C}(x) \wedge \mathrm{P}(y,x)) \vee \mathrm{C}(y)) \wedge \neg\mathrm{M}(a) \wedge (\exists x)(\mathrm{C}(x) \wedge \mathrm{P}(a,x)) \equiv$$

$$\equiv (\forall x)(\neg\mathrm{C}(x) \vee \mathrm{M}(x)) \wedge (\forall x)(\forall y)(\neg\mathrm{C}(x) \vee \neg\mathrm{P}(y,x) \vee \mathrm{C}(y)) \wedge \neg\mathrm{M}(a) \wedge (\exists x)(\mathrm{C}(x) \wedge \mathrm{P}(a,x)) \equiv$$

$$\equiv (\forall x)(\neg\mathrm{C}(x) \vee \mathrm{M}(x)) \wedge (\forall x)(\forall y)(\neg\mathrm{C}(x) \vee \neg\mathrm{P}(y,x) \vee \mathrm{C}(y)) \wedge \neg\mathrm{M}(a) \wedge (\exists z)(\mathrm{C}(z) \wedge \mathrm{P}(a,z)) \equiv$$

$$\equiv (\exists z)(\forall x)(\forall y)((\neg\mathrm{C}(x) \vee \mathrm{M}(x)) \wedge (\neg\mathrm{C}(x) \vee \neg\mathrm{P}(y,x) \vee \mathrm{C}(y)) \wedge \neg\mathrm{M}(a) \wedge \mathrm{C}(z) \wedge \mathrm{P}(a,z))$$

Per la skolemizzazione dobbiamo eliminare il quantificatore esistenziale, e a tale scopo è sufficiente sostituire un simbolo di costante, b, alla variabile individuale z da esso vincolata. Si ottiene la formula

$$(\forall x)(\forall y)((\neg\mathrm{C}(x) \vee \mathrm{M}(x)) \wedge (\neg\mathrm{C}(x) \vee \neg\mathrm{P}(y,x) \vee \mathrm{C}(y)) \wedge \neg\mathrm{M}(a) \wedge \mathrm{C}(b) \wedge \mathrm{P}(a,b))$$

alla quale resta associato lo schema di clausole

$$\{\{\neg\mathrm{C}(x), \mathrm{M}(x)\}, \{\neg\mathrm{C}(x), \neg\mathrm{P}(y,x), \mathrm{C}(y)\}, \{\neg\mathrm{M}(a)\}, \{\mathrm{C}(b)\}, \{\mathrm{P}(a,b)\}\}\,.$$

L’universo di Herbrandt è finito, e consiste nei due simboli di costante a e b che dobbiamo assegnare alle variabili individuali x e y in tutti i modi possibili. Si ottiene il seguente insieme di clausole:

$$\{\{\neg\mathrm{C}(a), \mathrm{M}(a)\}, \{\neg\mathrm{C}(b), \mathrm{M}(b)\}, \{\neg\mathrm{C}(a), \neg\mathrm{P}(a,a), \mathrm{C}(a)\}, \{\neg\mathrm{C}(a), \neg\mathrm{P}(b,a), \mathrm{C}(b)\},$$
$$\{\neg\mathrm{C}(b), \neg\mathrm{P}(a,b), \mathrm{C}(a)\}, \{\neg\mathrm{C}(b), \neg\mathrm{P}(b,b), \mathrm{C}(b)\}, \{\neg\mathrm{M}(a)\}, \{\mathrm{C}(b)\}, \{\mathrm{P}(a,b)\}\}\,.$$

Possiamo applicare l’algoritmo di Davis-Putnam!

Le clausole $\{\neg\mathrm{C}(a), \neg\mathrm{P}(a,a), \mathrm{C}(a)\}$ e $\{\neg\mathrm{C}(b), \neg\mathrm{P}(b,b), \mathrm{C}(b)\}$ devono essere soppresse perché sono tautologie; la clausola $\{\neg\mathrm{C}(a), \neg\mathrm{P}(b,a), \mathrm{C}(b)\}$ può essere soppressa perché contiene l’altra clausola $\{\mathrm{C}(b)\}$. Siamo dunque ricondotti a considerare il seguente insieme di clausole:

$$\{\{\neg\mathrm{C}(a), \mathrm{M}(a)\}, \{\neg\mathrm{C}(b), \mathrm{M}(b)\}, \{\neg\mathrm{C}(b), \neg\mathrm{P}(a,b), \mathrm{C}(a)\}, \{\neg\mathrm{M}(a)\}, \{\mathrm{C}(b)\}, \{\mathrm{P}(a,b)\}\}\,.$$

Pivot $\mathrm{M}(a)$:

clausole non contenenti né $\mathrm{M}(a)$ né $\neg\mathrm{M}(a)$: $\{\neg\mathrm{C}(b), \mathrm{M}(b)\}$, $\{\neg\mathrm{C}(b), \neg\mathrm{P}(a,b), \mathrm{C}(a)\}$, $\{\mathrm{C}(b)\}$, $\{\mathrm{P}(a,b)\}$;

$\mathrm{Ris}_{\mathrm{M}(a)}(\{\neg\mathrm{C}(a), \mathrm{M}(a)\}, \{\neg\mathrm{M}(a)\}) = \{\neg\mathrm{C}(a)\}$;

$$\{\{\neg\mathrm{C}(b), \mathrm{M}(b)\}, \{\neg\mathrm{C}(b), \neg\mathrm{P}(a,b), \mathrm{C}(a)\}, \{\mathrm{C}(b)\}, \{\mathrm{P}(a,b)\}, \{\neg\mathrm{C}(a)\}\}\,.$$

Pivot $\mathrm{C}(b)$:

clausole non contenenti né $\mathrm{C}(b)$ né $\neg\mathrm{C}(b)$: $\{\mathrm{P}(a,b)\}$, $\{\neg\mathrm{C}(a)\}$;

$\mathrm{Ris}_{\mathrm{C}(b)}(\{\neg\mathrm{C}(b),\mathrm{M}(b)\},\{\mathrm{C}(b)\})=\{\mathrm{M}(b)\}$;
$\mathrm{Ris}_{\mathrm{C}(b)}(\{\neg\mathrm{C}(b),\neg\mathrm{P}(a,b),\mathrm{C}(a)\},\{\mathrm{C}(b)\})=\{\neg\mathrm{P}(a,b),\mathrm{C}(a)\}$;

$$\{\{\mathrm{P}(a,b)\},\{\neg\mathrm{C}(a)\},\{\mathrm{M}(b)\},\{\neg\mathrm{P}(a,b),\mathrm{C}(a)\}\}\,.$$

Pivot $\mathrm{C}(a)$:

clausole non contenenti né $\mathrm{C}(a)$ né $\neg\mathrm{C}(a)$: $\{\mathrm{P}(a,b)\}$, $\{\mathrm{M}(b)\}$;

$\mathrm{Ris}_{\mathrm{C}(a)}(\{\neg\mathrm{C}(a)\},\{\neg\mathrm{P}(a,b),\mathrm{C}(a)\})=\{\neg\mathrm{P}(a,b)\}$;

$$\{\{\mathrm{P}(a,b)\},\{\mathrm{M}(b)\},\{\neg\mathrm{P}(a,b)\}\}\,.$$

Pivot $\mathrm{P}(a,b)$:

clausole non contenenti né $\mathrm{P}(a,b)$ né $\neg\mathrm{P}(a,b)$: non ce ne sono!

$\mathrm{Ris}_{\mathrm{P}(a,b)}(\{\mathrm{P}(a,b)\},\{\neg\mathrm{P}(a,b)\})=[\,]$;

$$\{\{\mathrm{M}(b)\},[\,]\}$$

Poiché abbiamo ottenuto la clausola vuota, l’insieme di clausole considerato non è soddisfacibile; dunque la (iv) è conseguenza logica delle premesse.

Se fossimo in una lezione effettiva, “dal vivo”, a questo punto mi aspetterei un vostro intervento. Qualcuno secondo me dovrebbe chiedere: perché il concetto “padre di” è stato espresso nel linguaggio di logica dei predicati attraverso un predicato binario e non, come sarebbe naturale, attraverso una funzione?

Vabbè, probabilmente nessuno l’avrebbe chiesto comunque, ma se ci ripensate un attimo l’osservazione è davvero molto sensata e merita una risposta. Sì, è vero che il modo “naturale” (e anche oggettivamente più semplice) per esprimere la paternità è quello di introdurre una funzione p_0 che a ogni personaggio del ragionamento associa il padre. Però ci imbarazzerebbe tradurre la conclusione: infatti “Alf non è il padre di alcun cane” dovrebbe venire espressa con una formula tipo “$\neg(\exists x)(\mathrm{C}(x)\wedge(a=p_0(x)))$” dove per poter andare avanti non basterebbe esprimere l’uguaglianza con un predicato binario: avremmo bisogno di poter utilizzare le proprietà intrinseche dell’uguaglianza, che significa (ricordate?) “essere la stessa cosa” e quindi richiede di poter sostituire dovunque uno dei due termini dichiarati fra loro uguali con l’altro. Insomma, dovremmo usare regole della logica ***più ampie*** di quelle che abbiamo studiato noi.

Eh, sì, a questo punto devo gettare la maschera… La logica dei predicati che avete visto in questo insegnamento non è ***tutta*** la logica dei predicati (ma del resto nemmeno l’algebra che avete visto in questo insegnamento è tutta l’algebra, e nemmeno la teoria dei grafi che avete visto in questo insegnamento è tutta la teoria dei grafi!). Esistono tecniche per portare avanti l’analisi anche quando l’universo di Herbrand è infinito (cercate con Google “algoritmo di unificazione”… e buona fortuna!), ed esiste un particolare trattamento dell’uguaglianza, che permette di gestire le funzioni e la loro traduzione (mediante simboli di funzione) nel linguaggio di logica dei predicati adottato. Ma tutto questo esula dal programma di questo insegnamento (e, non mi vergogno ad ammetterlo, anche dalla mia competenza specifica).

Nella prossima ora faremo un altro esempio che credo interessante, ma sempre evitando di introdurre la formalizzazione di funzioni.